Request for Impacted Status: 
Department of Computer Science

Executive Summary

In the last 5 years the number of Computer Science proposed majors has increased from 238 to 1437, and the number of CS declared majors has increased from 143 to 612 (see Figure 1). This is a six-fold increase. In 2015-16, CS had 18.5 permanent faculty, which corresponded to a 78:1 CS majors to faculty ratio. The campus instructional workload ratio for that same period was 26.1 student FTEs to faculty payroll FTEs while CS was at a high 42.8. Even more extreme ratios in subsequent years are likely if no action is taken. The CS department has done all that it can to support the surge in enrollment due to the rapidly rising interest in computer science without a declaration of impaction status. The measures taken, detailed later, included precipitously increasing class sizes, hiring more lecturers, requesting augmented teaching resources ranging from temporary FTEs to teaching assistants and graders, and streamlining the curriculum in ways that have been fairly draconian. As the surge in interest accelerates exponentially, however, these measures are falling short. It is becoming increasingly difficult for CS majors to enroll in the classes that they need to graduate in time, to the point that in some quarters several students have been unable to enroll in any classes needed for their CS major and are considering dropping out or taking periods of leave. This trend will only get worse as the imbalance between CS student population and CS faculty grows.

The good news consists in the existence of admission criteria into the CS major that are guided by student success considerations and which, if adopted, would lead to a very minor reduction in the number of students who successfully graduate, while steeply reducing the number of students who fail to graduate. The underlying fact is that the great majority of proposed CS majors students who fail to declare a CS major do not migrate to other majors within UCSC: rather, they leave the university, often carrying shattered dreams and student debt. If we could admit students to the CS major following these success-based criteria, students success for students admitted to the major would increase, as they would have more teaching resources available to them.

Therefore, we are asking for the following:

- Immediate declaration of Impacted Status for the CS BS and BA programs.
- Insertion into the UCSC application and admissions documents, literature, and web pages of a statement to the effect that the choice of major may be taken into account in the admission process for UCSC.

With the above framework in place, we will then be able to fine-tune the specific admission criteria and major declaration policies. In particular, our department will engage with senate committees, other departments, and the administration to:

- Develop the precise admission criteria and target numbers for admitting students to UCSC in the CS major. The targets will increase tracking the increase in our faculty size and teaching resources. The goal is to have the admissions process for the CS major in place for the Fall 2017 application season; hence the need for inserting now the above
statement in application material.

- Develop policies and procedures regarding the migration to the CS major from other majors, limiting it to particularly meritorious cases.
- Ensure that enrollment into CS classes is restricted or prioritized to students with CS major, along with students of majors with which CS has agreements to share classes (such as CE, TIM, and CM).

![Figure 1: Number of yearly CS major declarations (blue), and numbers of total enrolled CS declared (red) and CS declared or proposed (green) majors, 2010 to 2016.](image)

We will develop the relevant policies for admission and transfer to the major in collaboration with the relevant senate committees, keeping into account also the impact on the diversity of the major student population. As discussed later in more detail, we are aware that adopting stricter admission criteria might impact the diversity of the incoming students, but we believe (and there are indications on other campuses) that a better student-to-faculty ratio leads to better diversity outcomes (major declaration, and graduation) for the students that are admitted.

In the longer term, we expect that the CS teaching resources will grow, including the number of tenure-track FTEs. As our resources will grow, we will periodically re-examine the admission targets and criteria, to ensure the department can serve the largest number of student, while ensuring the best possible outcome for the students in the major.

The consequences of inaction are steep. If no modified admission policy is implemented, then the only recourse left to ensure that CS declared majors have sufficient resources to graduate will be to increase the difficulty of declaring the CS major once students are on campus, via
more challenging classes, stricter criteria, or both. This will result in large numbers of students who are admitted but are then unable to declare a CS major, and who will likely leave UCSC at a great personal cost in time and money.

The Growth in the CS Student Body

Figure 1 illustrates the growth in our student body. In spite of the several measures we have taken, and which are detailed later, this exponential growth is outpacing our teaching capacity. We have had to steeply increase class sizes, modify classes in ways that are not beneficial to students (for instance, simplifying assignments, and doing without project components). Yet, students are experiencing more and more difficulties getting into the classes they need to graduate on time. As the surge in declarations will work through the system (see Figure 1 again), we expect the situation to grow worse before it can get better.

In the eyes of the students

To better realize the situation of the CS major it is useful to take a look at some typical classes, and at comments by students seeking permission codes. The comments have been edited only by removing the student’s names, to preserve their privacy.

CMPS 121 (Mobile applications). This class went from 57 students in 2012, to 230 students in 2017 with a waiting list of 70. 2016 was the last year in which every student could do a project; in 2017, project are by instructor approval, and we plan to accept about 30 of them, involving about 80 students, or a third of the students enrolled. These are comments from students who are requesting permission codes:

“ I am currently a third year undergrad student at UCSC majoring in CMPS B.S. Sadly my enrollment time is tomorrow at 9 am and every upper division class that I can possibly take to further my education for my major is full at the moment. I’m not really sure exactly what to do at the moment as I can’t take any classes for my major but I thought I would go ahead and ask if you had any power to help me into your class.”

“It's the only CS-related class that has a waitlist less than 60 people and it's possible for me to take in this spring.”

“At the moment, I will not be able to graduate on time because of the situation. I want to go into the mobile applications field after graduation and want to use this class as my upper div elective. Also there are no other upper div CS classes available for me to take as they are all full so I can’t continue the CS major.”

“Since I came to UCSC, I have always faced difficulty in getting any Computer Science upper-division classes. Even with junior year standing, CS 121-01: Mobile Applications was already full when I was allowed to enroll for my classes for the Spring 2017 academic quarter. I would be extremely thankful for a permission code because I want to focus on Mobile Application Development. I want to intern or work for companies such as Appstem, King (Activision Blizzard), Snap, or Instagram. I am going in my third academic school year yet I am still facing stiff competition amongst the increasing population of Computer Science non-transfer and transfer students.”

“I need this class in order to satisfy the 2 CGE pre-requisite for the capstone course in
Game Design. If I cannot get my second CGE then I cannot take my 3 quarter capstone sequence, and will not be able to graduate on time, and will have to wait a whole year to take it in my 5th year.”

CMPS 183 (Web Applications). This capstone class went from 70 students in 2012, to 200 students in the Fall section 2016 alone, along with a waitlist of 50-60 students, and more students in the Winter section of 2016. 2016 is likely the last year in which every student did a project; in the future, the only way to run the class will be to have projects be by instructor permission, with considerable loss in the learning experience. Here are some comments from desperate students trying to get a permission code:

“I really wanted to take web applications courses, since I was in high school. I want to build websites in future. I waited for a long time to perfect my skills and complete required courses to able to take it. But now its seems I won’t be able to add this course. I tried to add this course last year and now I don’t want to lose this time too.”

“I am planning on graduating this Fall, and this is the last class that I need to take to graduate. Not taking this class in the Fall will cost me thousands of dollars in tuition which will come out of my own pocket. “

“I need a permission code so I can fulfill my capstone requirement and be on track to graduating by the end of the year!”

CMPS 185 (Technical Writing and Communications in CS). This course was developed to help students meet their DC requirement. Originally it had an enrollment cap of 35; enrollment has now swollen to nearly double that (58), and yet had an extensive waiting list. Here are statements from students seeking a permission code:

“I am a fourth year CS transfer student on track to graduate at the end of spring quarter 2017. I had my enrollment appointment for winter quarter yesterday in which I attempted to enroll in you upcoming CMPS 185 course, however it filled up before my registration slot even opened. Besides this 185 class, there are currently no other course options for DC requirement in the coming winter and spring quarters. I am hoping that there is a possible resolution and course of action that I can take to allow me to graduate on time.”

“I’m a computer science senior hoping to graduate this spring, but I didn’t get into CMPS 185, which I need for graduation. It doesn’t look like it will be offered again this year, so I’m really feeling stuck. I really don’t want to spend another $10k or so to pick up the DC requirement next year, so I was wondering if I could get a permission code.”

“I am currently a senior at UCSC, and am studying computer science. I wanted to take your Technical Writing class next quarter (Winter 2017). Do to an incredibly hectic work schedule and a family emergency, I missed my enrollment appointment by 5 hours and the class was closed by the time I was logged on to the portal. I am unable to find any other electives that work with my schedule, and was wondering if there was any way you could open a spot for me.”

“I’m currently on track to graduate this year and have unfortunately not made it into your CMPS 185 class which is necessary for me to graduate. I’m on the wait list as position 17 as of the moment and would like to know if there is any possible way in securing my position into class. I know I could wait until the end of the wait list period but I’ve had
many quarters in the past where this has not worked out to my benefit.

“I would love the opportunity to take your CMPS-185 class in the Winter 2017. I'm currently high (in the 20’s) on the waiting list, and it is one of the classes I need to fulfill my DC requirement...I tried getting other DC requirements but was unable to register at all for any of them and the seats all got taken. Is there any way there will be more spots opening up? Or another section being taught this quarter or next? I really don’t want my graduation delayed so long... It’s so hard getting CS courses here.”

“I am a Computer Science BS Undergraduate who is in his senior year. I am expecting to graduate in Spring 2017, however, I still need to fulfill my DC requirements. Looking at the UCSC Computer Science Course Web Page, it seems like CS 185 is the only class that is offered for the remaining of the year that also fulfills the DC requirement. I was planning to take CS 185 in Winter 2017, however, the class was full before I was even able to sign up.”

Why are CS student numbers going up?

The biggest long-term reason for the increase in our student pool is likely demand from industry, as well as the ever-growing role of information technology in every industry and production sector. Fields that once relied on labor, such as agriculture, now use sophisticated software to monitor fields, automation and drones to survey and spray them, and more technology to harvest them, package the products, and distribute them. Even white-collar fields such as paralegals are experiencing the effects of the inroads of automation and information technology. As this happens across fields, prospective students take notice, and strive to maximize their computer science expertise.

In Figure 2, we compare the growth of the SOE CS and CE departments1 with the average growth of departments participating in the 45th CRA Taulbee Survey2. The CRA Taulbee Survey is a survey of PhD-granting CS, CE, and Information departments in the United States and Canada; in 2016 the survey solicited data from 266 departments, of which 178 responded. Figure 2 shows that the growth of the number of students in the UCSC CS department tracks very closely the average growth of departments with at least 25 tenure-track faculty, while the growth in CE students tracks closely the growth of departments below 25 tenure-track faculty. This shows that the growth in number of students is a national trend that shows no sign of stopping.

1 The data from SOE is from Infoview/Student Universe/Major/SOE-historical 3 QTR average.  
An influential study that documents this phenomenon is the STEM report issued by Georgetown University's Center on Education and the Workforce\textsuperscript{3}. This survey laments the shortage of STEM (Science, Technology, Engineering, and Mathematics) majors -- a common refrain these days -- and, interestingly, predicts that by 2018 more than half of STEM jobs will be for computer specialists. University of Washington's President Ana Mari Cauce talked about the increasing demand for CSE degrees from students, saying that she is not worried about potential changes in the job market because CSE degrees are "very flexible": "I am not a gambling person, but this is a sure bet", said Cauce in an interview to GeekWire in 2017. Last year, CSE became the leading first-choice major among confirmed incoming UW freshmen. UW says it currently has to turn away two out of three qualified student applicants to the CSE department.

Additionally, the CS programs at our sister campuses either tend to reside in Engineering Schools that control the number of CS students admitted or have already been declared impacted. Thus students interested in CS are forming an increasing fraction of our applicant pool, and tend to have a higher conversion rate (admission to SIR to enrollment).

The startling pace of growth in our numbers might evoke the "dot-com" boom of the late 1990s, when CS enrollments also rose. One might thus wonder whether we are experiencing a similar short-term phenomenon. While it is difficult to predict the future, the growing role of information

\textsuperscript{3} Available at: http://www9.georgetown.edu/grad/gippi/hpi/cew/pdfs/stem---_complete.pdf
technology in our society is a long term trend that will continue, and with it, we expect the growth in popularity of STEM and CS to also continue. Quite simply, humans are going from doing jobs themselves, to teaching machines how to do jobs for them, and the latter involves many CS skills, from programming, to machine learning, to data organization, in fundamental ways.

**Student Pathways in the Major**

![Figure 3: Status of Fall 2014 proposed CS majors in Fall 2016.](image)

Thanks to data made available by John Tamkun through the UC Davis Ribbon Tool, we are able to analyze the Fall 2014 cohort’s transition from proposed major to declared major. The data is depicted in Figure 3. The upshot, insofar as our impaction analysis is concerned, is that the great majority of students who were CS-proposed in 2014 were either in CS (proposed or declared) in 2016, or had left the campus. Successful migration to other majors was only 14%, of which 4.7% in SOE and 9.3% outside. Thus, CS is not a main intake channel for majors outside CS, and limiting the number of students admitted to CS will not negatively impact the students that flow to other campus majors.

In detail, of the 236 proposed majors entering Fall 2014, 109 (46%) declared the CS major by
Fall 2016. Another 39 students (17%) left the campus, while 36 students (15%) continued as proposed majors (the data is unclear as to whether these students are in the process of declaring the major or on their way to leaving). Together, these students account for 78% of the students entering with the intent to study Computer Science. Of the 33 students (14%) that declared another major, 11 (4.7%) declared a SOE major (1 Bioinformatics, 3 Computer Games, 2 Robotics, 3 TIM, 2 CE), and 22 (9.3%) declared another major (6 PBSci, 2 Humanities, 4 Arts, 10 Social Sciences). The remaining 8% were still proposed majors, 7 (3%) in non-CS SOE majors, and 12 (5%) in non-SOE majors.

In addition to the frosh coming in as proposed majors, an additional 38 students declared the CS major (22% of the CS major declarations). Most (29) entered as undeclared students, with the remainder entering as proposed majors in PBSci (6 students) and Social Sciences (3 students). After two years, 24 students migrating into our program were still proposed majors. Again, the largest group (9 students) entered as undeclared, with another 8 coming from PBSci, 4 from Social Sciences, 2 from Arts, and 1 from Humanities. The total migration into our program (62) exceeded the number of students migrating into other campus programs (52).

**Accommodating Growth**

The department has undertaken several efforts to accommodate the growth in enrollments. We present a long list of steps we have taken in Appendix 2; here, we summarize some approaches, and offer comments on others that we consider inappropriate.

**Use of non-ladder faculty to offer courses, and increasing offerings**

We are making extensive use of lecturers, and we are aggressively soliciting graduate student instructors. We have pushed this approach likely beyond its appropriate and reasonable limits: in 2015-16, faculty performed only 32% of undergraduate teaching, with the rest being done by lecturers. This carries two problems:

- This is clearly not what undergraduates think they have signed up for when they elect to attend UCSC.
- The lecturer pool is an unstable resource, with lecturers joining it and leaving it annually. Recruiting lecturers has been problematic due to the non-competitive compensation compared to industry alternatives.

In summary, 68% of undergraduate teaching is built on the shifting foundation of an ever-changing lecturer pool, whose availability cannot be taken for granted from one year to the next. To this must be added the heavy burden required in mentoring each year the new lecturers, and ensuring their availability.

**Why not offer courses more often, including during summer?** While we have encountered challenges hiring instructors in the summer we try to offer courses highly impacted during the school year in the summer. We have seen our summer enrollments increase three-fold since 2012. Furthermore, we plan to offer required courses every quarter, and we plan double offerings of CMPS 12B and 101 next year. However, our advanced courses require special

---

4 The percentages have been computed by considering the total enrollment in undergraduate classes taught by lecturers vs ladder faculty.
expertise and we are unable to find additional qualified lecturers, and we can ill afford increasing even further the share of teaching done by lecturers. In summer, many students pursue internships, reducing the availability of qualified GSIs and TAs, but we do what we can.

**Increase in class sizes**

We have let our classes get progressively larger. We routinely request larger rooms as enrollments dictate, and some classes are at four times their pedagogically determined size limits. We are also making widespread use of technology to cope with larger classes, for instance:

- Several of our classes are recorded, to accommodate an enrollment higher than room capacity
- Several classes are using peer-grading tools such as CrowdGrader, in which students grade each other's work, to be able to assign non-trivial homework in spite of limited TA resources for the class size.

**Use of technology and online education**

The department is making large use of technology to be able to educate more students more effectively. We present a more comprehensive list in appendix; our efforts include:

- **Use of recorded lectures** to allow more people to follow a class than can fit in a classroom. This is the educational equivalent to over-booking a flight, in the knowledge that not all students attend all the time, and we pioneered this even when the approach was not endorsed yet by the campus.
- **Use of peer-grading tools** such as CrowdGrader. Peer grading enables instructors to assign complex homework (ranging from writing assignments, to programming assignments and projects), which is then graded by the students themselves. This has both an educational goal (student learn from the work of others), and a scaling goal, as it becomes possible to assign such homework with appropriate frequency even in large classes. The CrowdGrader tool was born in our department, and is now used in several sections of large classes (5J, 121, 183, 185).
- **Use of automated grading tools.** In several classes, including 101, our faculty has developed methods for automatically grading programming assignments against benchmarks, again allowing us to better cope with scale.
- **Use of overflow rooms.** We have experimented with simulcasting lectures to an overflow room to accommodate more people than can fit in a single room. The ability of students in the overflow room to participate actively in the lecture and ask questions are factors that still need some work before we try this approach in more extensive ways.

One often-heard suggestion in the face of the enrollment increase is that we make more extensive use of purely online education. We think this is inappropriate, for several reasons.

First, it has been proven time and again that online education can work for strongly motivated and skilled individuals, but generally fails for weaker or less motivated students, where the lack of a social motivation (the peers taking the same physical class) is detrimental to learning. We worry that a move to increase online education would have damaging repercussions on diversity, in particular as we expect difficulties with first-generation students. Further, on-line courses would reinforce the stereotype that Computer Science is the domain of asocial "nerds"
who spend all day in front of their computers, a stereotype which is not helpful in achieving gender diversity.

Second, online education is not what the students bargained for when they enrolled in UCSC; if they wanted online classes, they could have taken them from Coursera or many non-UCSC alternatives at a fraction of the cost of attending UCSC. As a student succinctly put it, "I personally can't stand the idea of paying tuition for an online only class because there are so many free resources already online" (comment of Fall 2016 CMPS 183 student). The value of UCSC consists precisely in the personally delivered instruction and guidance.

Third, online classes require a large time investment to create; once created, they tend to remain static, due to the high cost of making changes. This results in a static curriculum which quickly gets out of step with the times. CS is a very dynamic field, and we update our CS classes on a yearly basis to keep them current; online education would result in staleness.

This said, we are making every effort to use technology to increase our teaching effectiveness and reach, as mentioned earlier. This includes recording lectures to overcome room capacity constraints (against campus policy), using peer-grading for assessing homework submissions, using automated evaluation methods for several programming assignments, and more (see the Appendix for a full list of steps).

**Curriculum reorganization**

The department has recently proposed some changes to CMPS BS degree requirement following the principle of "smaller core and more flexibility" in choosing upper division elective computer science courses without reducing the number of upper division electives. The changes benefit students by creating more paths for students to fulfill major requirements, and they are consistent with a desire of broadening of the knowledge areas in Computer Science following the guidelines described in the ACM and IEEE Computer Science Curricula 2013 as recommended by the CS external review committee. These changes are currently being reviewed by CEP. In particular, we recommend going from requiring all five of CMPS 102, 104A, 111, 112, 130, to just three cores: 102 (analysis of algorithms), 112 (comparative programming languages), and 111 (operating systems); students then need to take 2 upper division classes from a wider set of electives. If adopted, students will have more flexibility in choosing their electives, thus creating more paths for students to fulfill major requirements while acquiring a robust skill set for broader participation in the impact of computer science.

**Use of TAs and undergraduate tutors**

In previous years, having sufficient TAs has been a challenge. Many upper-level engineering classes are about “building stuff”: project-oriented classes where students need help and guidance about their individual work (as opposed to general homework). We often had to teach project-based upper-division classes with 1 TA for every 100 students. This was happening in part because the allocation model for TAs was based on past enrollments, thus penalizing classes whose enrollment grows steeply from one year to the next. Recently, we have received a more generous TA allocation, and we are looking forward to seeing the effects fully percolate in our classes; we expect it will bring considerable relief.

Our department has developed a methodology for successfully utilizing undergraduate tutors that constitute an essential part of the workforce of programming courses. We should be able to
scale the size of this pool with the number of students. The resource we cannot scale, however, is laboratory space.

**Updated CS Major declaration policy**

We are currently considering a modification in our major qualification policy. According to the new policy, within the first two years the students have to achieve a GPA of at least 2.8 in two separate groups of classes, one group being math classes, the other group being CS classes. This policy is stricter than the current policy, where we ask for a GPA of at least 2.8 for a single group of classes. This will help, as it will reduce somewhat the number of students that can declare the major. As we explain in more detail in the next section, however, preventing students who are already on campus from declaring the major is far from desirable.

**New Proposed Engineering major gating policy**

Under consideration, and likely to be approved by CEP, is a new Engineering major gating policy. Under the policy, in order to be eligible to declare a School of Engineering major, students must be listed as a proposed major in one of the following: Bioengineering, Bioinformatics, Computer Engineering, Computer Science: Computer Game Design, Computer Science (B.A. and B.S.), Electrical Engineering or Robotics. Students pursuing Network and Digital Technology and Technology and Information Management are not required to be proposed in these majors in order to declare. Under the policy, students in their fourth quarter or beyond who would like to become a proposed major, or students who are proposed and would like to retain their proposed status, must have passed Math 19A or 20A, and two additional BSOE classes from the following list within their first three quarters: CHEM 1A, CMPE 12, CMPE 13, CMPE 16, CMPS 11, CMPS 12A, CMPS 12B, MATH 19B, MATH 20B, PHYS 5A, or PHYS 5C. Students who are in an Engineering proposed major and who do not meet these criteria will be removed from their proposed major. Students who want to change to an Engineering proposed major must also meet these criteria. Students that do not meet these criteria can appeal this decision.

Preliminary study of this proposal, carried out by John Tamkun, chair of CEP, indicates that the policy is very effective in screening out students who have a very low probability of successfully graduating (well below 5%). Thus, the policy is effective in reducing the number of SOE majors, while only minimally affecting the chances of qualified students to graduate.

Again, it would be better to screen students even earlier, at application time. However, compared to the above CS Major declaration policy, this policy has the advantage that the screening occurs entirely in the first year, rather than possibly spanning the first two years.

Modeling done by John Tamkun, Chair of CEP, indicates that this policy will lead to a reduction in CS majors.

**Course withdrawal deadline**

In view of the long waiting lists for CS classes, we are considering bringing forward the deadline for withdrawing from a class, possibly to the end of the first week of classes. In this fashion, students on the waiting list would be able to better profit from withdrawals.

**What could still be done?**
We argue in this proposal for a modified and success-inspired acceptance policy for CS majors, for selecting students who have real chances of succeeding, and for focusing our teaching and mentoring resources on them.

The steps that remain to be tried, and that can be implemented without an impaction policy, are not pleasant. If it is not possible to select students before they come to campus, the remaining solution will be a steeper selection of the students once they are here, via more grueling and strictly graded coursework, higher bar for declaring the major, and so forth. These steps will result in a large waste of resources on campus part, as efforts will be spent on students who have few chances to succeed. But more importantly, the consequences will be disastrous for those students. As we have seen from Figure 3, those students will by and large leave UCSC, and they will leave at a steep financial loss (and likely with student debt), with shattered dreams, and after wasting some of their most crucial formative years in a fruitless effort that could have been better directed. The total human cost would be staggering.

**Limits to growth**

In spite of our efforts, we are encountering limits to growth in several directions.

**Lab space.** Many of our classes require computing labs, and these are filled at capacity. We have converted to teaching labs all of the spaces we could, and we do not know where it will be possible to find more space for these. We are trying to counter this by requiring laptops, but beyond providing “computers”, labs are useful as they allow students and instructors to interact while working on the code. We also observe that relying entirely on laptops is creating difficulties for our less financially able students, whose laptops are often markedly inferior to the ones of their peers; in this sense, labs are welcome equalizers.

**Project classes.** Some of our classes involve projects; this is especially common in the upper-division classes that can serve as capstones and lead to graduation. We have tried to have projects in classes of up to 200 students, and it was not pedagogically successful. 200 students translated to 80 projects, and it was simply impossible for the instructor and the two TAs to follow all of these projects in a meaningful way. Even scheduling the project presentations was problematic, requiring 12 hours of large classroom reservations spread over two days of finals week.

**Office hours.** Larger classes mean more students to office hours. While TAs allocations are now more closely tracking enrollment, students still need to be able to talk to instructors. This is the only way in which students can get career advice, can get involved in undergraduate research, and can get advice for their own projects. We are making wide use of technology, such as online Piazza forums. While Piazza helps broadcast the answer to one student to all the potential students who might be facing the same issue, it does not help instructors provide more individual guidance and advice (such as career advice, or project advice) to particular students. In addition, the lack of sustained contact with individual students prevents instructors from writing the kind of insightful recommendation letters needed for admission to the top graduate schools.

**Balance of undergraduate student to faculty ratio.** With 1437 majors and 22.5 total payroll faculty (regular and temporary), our student-to-payroll ratio is 64. We do not expect to be able to achieve an oft-quoted ratio of 20:1 anytime soon, but 64:1 is exceptional. Because of the exponential growth in demand for Computer Science, even if admissions plateau this coming year, our major will continue to grow dramatically, exceeding 2000 in four years (an 89:1 ratio).
**Student guidance.** In the final year, and before, student benefit from interaction with faculty in the classes, and in the advising sections, in order to refine their skills, get involved in interesting projects and research, and so forth. UCSC prides itself to have a participatory undergraduate experience, where the best undergraduate students can work alongside graduates and faculty in research projects. This is becoming a possibility for fewer and fewer students, as the student to faculty ratio increases. The following table shows our class size distribution in 2015-16 compared to the UCSD CS department’s class sizes when they were granted impacted status.

<table>
<thead>
<tr>
<th>Class size</th>
<th>UCSD</th>
<th>UCSC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-99 students</td>
<td>69 (58%)</td>
<td>30 (54%)</td>
</tr>
<tr>
<td>100-149</td>
<td>26 (22%)</td>
<td>11 (20%)</td>
</tr>
<tr>
<td>150-199</td>
<td>22 (18.5%)</td>
<td>5 (9%)</td>
</tr>
<tr>
<td>200+</td>
<td>2 (1.5%)</td>
<td>9 (16%)</td>
</tr>
</tbody>
</table>

**Student Advising Imbalance.** Student advising is a crucial part of department operations. There are a large number of students who are curious about our major and are potentially interested in transferring in. Many need advice on which pathway through the curriculum to take, given their particular interests. There are some who are concerned about their performance in certain classes and want clarification about the repercussions it will have. And so on. This counseling is essential and cannot be done outside the department. The enormous growth in the undergraduate body over the past two years has placed major strain on the SOE advising office.

**Proposal for Impaction Status and Consequent Actions**

Our goal in this proposal is to seek impacted status for the CS major, and to work towards a set of policies, including as central point a new admission policy to UCSC for the CS major, that will limit the number of students who enter the major.

We propose that the new policy be developed in such a way as to act on admissions to UCSC for the CS major, rather than on the mechanism by which students can declare or stay in the CS major once on campus. Admitting students interested in CS to UCSC, while preventing them from later pursuing the major, is a tragic waste of resources and human capital. As we discussed earlier, students admitted to UCSC as proposed CS majors, and who fail to progress in the major, by and large leave the university. But they leave with broken dreams, often piles of student debt, and after wasting some of their most crucial years. These students would have fared much better if we declined admission, so that they could have instead considered career or education paths where they might have met with more success. On our end, these students use advising and TA resources that could be better spent on students that have more possibilities to be successful.

**Analysis**

**Success.** An analysis performed by John Tamkun, Chair of CEP, indicates that it is likely possible to modify the admissions criteria in a way that selects students most likely to succeed, while leaving out few students who would have been successful. For instance, Figure 4 reports...
the correlation between Math SAT score and outcome in various classes for students entering in Fall 2014, following the classes they have taken until their junior year. We see a strong correlation between Math SAT and outcome in the foundational CS classes (5J, 11, 12A, 12B); few of these students make it to CMPS 101 by their junior year, as indicated from the size of the bubbles for the various quartiles. This indicates that it should be possible to refine admissions criteria in a student success oriented way, limiting the number of students that apply to the CS major with few chances of a good outcome.

Figure 4: Course enrollments and grades, entry in Fall 2014 to junior year, disaggregated by Math SAT score quartiles. Violet is the top quartile, red is the second (from top), green is the third, and orange is the bottom quartile. The area of a bubble is proportional to the number of students.

In another look at the same data, in Appendix 1, Figures 5-6, we report a comparison between the destinies of students in the top three Math SAT quartiles, and in the bottom quartile. The difference in student success (here measured as “SOE-Declared”, that is, the students who were able to declare a SOE major) is evident. Figure 7 shows the success of the students who are in the top three quartiles of Math SAT, and who meet the proposed SOE major gating proposal described earlier. These are high success-rate students on which we would focus our teaching resources, if both a top-3 quartiles Math SAT admission and the SOE gating proposal were in effect.

We do not necessarily advocate using the top-3 Math SAT quartiles as admission criteria. Rather, we propose to work together with the Office of Admissions, CEP, CAFA, and committees interested in success and student diversity, to develop precise admissions criteria to be used.

Diversity. One concern in adopting a stricter admissions policy for CS majors is that this might negatively impact diversity. But the diversity that truly matters is the diversity in the students who succeed to graduate, not in the ones that are admitted. Indeed, as noted above, admitting students who are unlikely to graduate is likely to have negative impacts on these students, both financially, in their use of time and motivation, and because they might have pursued more successful options elsewhere. Indeed, our Dean for Undergraduate Students Larrabee reports that at UCLA, stricter admissions criteria were initially tied to fewer minority admissions, but subsequently to more minority graduations. In our School of Engineering, we have had good results in supporting minorities via our MEP program (see https://mep.soe.ucsc.edu/), to the point where MEP-supported students achieve successful graduation at rates higher than most other groups. Intensifying such programs, and focusing them on students who are more likely to succeed, would be beneficial.
The Impaction Request

In view of the above analysis, we are asking for the following:

- Immediate declaration of Impacted Status for the CS BS and BA programs.
- Insertion into the UCSC application and admissions documents, literature, and web pages of a statement to the effect that the choice of major may be taken into account in the admission process for UCSC.

With the above framework in place, we will then be able to fine-tune the specific admission criteria and major declaration policies. In particular, our department will engage with senate committees, other departments, and the administration to:

- Develop the precise admission criteria and target numbers for admitting students to UCSC in the CS major. The targets will increase tracking the increase in our faculty size and teaching resources. The goal is to have the admissions process for the CS major in place for the Fall 2017 application season; hence the need for inserting now the above statement in application material.
- Develop policies and procedures regarding the migration to the CS major from other majors, limiting it to particularly meritorious cases.
- Ensure that enrollment into CS classes is restricted or prioritized to students with CS major, along with students of majors with which CS has agreements to share classes (such as CE, TIM, and CM). Other requests will be handled via permission codes, so that special cases for meritorious students can be made.

In the longer term, we expect that the CS teaching resources will grow, including the number of tenure-track FTEs. As our resources will grow, we will periodically re-examine the admission targets and criteria, to ensure the department can serve the largest number of student, while ensuring the best possible outcome for the students in the major.
Appendices

Appendix 1: Success and Math SAT performance

*Figure 5: Student success (accomplishment of SOE-Declared in Fall 2016) for the top three quartiles of Math SAT score applicants.*
Figure 6: Student success (accomplishment of SOE-Declared in Fall 2016) for the bottom quartile of Math SAT score applicants.
Figure 7: Student success (accomplishment of SOE-Declared in Fall 2016) for the students in the top three quartiles of Math SAT score, and who also meet the proposed SOE major gating proposal.
Appendix 2: List of Measures Taken to Cope With Increased Enrollment

The CS department has taken many steps in order to cope with the increased enrollment. Below is a detailed, and yet likely not exhaustive, list of measures taken.

Increasing our ability to offer courses:
- Aggressively soliciting the use of GSI (graduate student instructors) to deliver lectures.
- Recruiting international lecturers.
- Efforts to recall emerita.
- Teaching upper-division courses in conjunction with a graduate course overload that share lectures.
- Voluntary course overloads by some faculty.
- Response to unexpected student demand by scheduling additional sections, often on an emergency basis.
- Increased usage of lecturers to potentially unhealthy levels.

Programmatic and Policy Changes:
- Allowing capstone classes to grow (this quarter one already has 230 students)
- More than doubling the sizes of DC courses.
- Requesting enrollment priority for majors and SOE students in most classes.
- Increasing class sizes (general data?).
- Modified MS requirements to encourage increased use of “mezzanine” upper division courses by MS students in an attempt to conserve on graduate offerings.
- Controversial changes in degree requirements to replace particular course requirements with electives.
- Creating added DC capacity through “add-on” 2-unit extensions of existing classes that are taught as an overload on the regular faculty teaching load.
- Encouragement of summer offerings, although finding qualified and willing instructors remains a problem.
- Formation of a Departmental Teaching Committee that archives course materials, mentors lecturers, promotes consistent grading policies, and disseminates best practices for effective instruction in large classes.

Resource Utilization:
- Since our enrollment is dynamically growing, we have made efforts to get more information to better project future enrollment for each class, but thus far these have not been satisfactory.
- Aggressive use of tutors in lower division courses, extending the available TA resources.
- Managing enrollment by reserving seats to be allocated by a petition process allowing enrollment by those students with the most critical need. Since the SOE staff allocated to our program is already overburdened, this process (dealing with 100’s of petitions a quarter) was developed, and is managed and executed exclusively by regular faculty.
- Requesting larger classrooms in the planning process, including making frequent use of the largest rooms on campus.
- Frequent (at least 5/quarter) enrollment-driven requests for larger classrooms and having our schedule perform extensive juggling of classrooms to maximize their utilization.
• Forcing lecturers at less than 90% time to share offices.
• Use of TAs in very large graduate courses, reducing need for additional graduate offerings.
• Discouragement of moneymaking extension enrollments to preserve capacity for our on-campus students.

Teaching Adjustments:
Different instructors may adopt different adjustments, often depending on the course being taught, and many of them result in a diminished student experience.
• Reliance on group homework rather than individual homework. Although group interactions can improve the learning experience, the lack of feedback on student’s individual progress can be problematic for the less-prepared students.
• Fewer and less extensive graded assignments.
• More concretely delineated assignments as opposed to open-ended ones.
• Dropping term projects or making them optional.
• Changing exams from essay/derivation format to short answer.
• Use of peer grading such as the CrowdGrader system developed by our faculty.
• Elimination of student presentations in almost all classes.
• Use of secondary overflow rooms with telecasting. Students in the secondary rooms usually cannot participate in the class discussion on the same levels as the students in the primary room.
• Use of multiple versions of exams and quizzes, and other efforts to identify and reduce academic honesty violations.
• Making sections optional rather than required: this is easier to manage in large classes and results in smaller group of more engaged students, but results in many of the students who would benefit from sections failing to attend.
• Additional training and coordination for the larger staff (TA’s and graders) required for larger classes.

Appendix 3: Questionnaire on Coping with Enrollment

We report the answers to questions on strategies for coping with increased enrollment. In the following, we use the following legend:

A: Can be performed by any department at any time
B: Requires CEP approval, routed through Dean’s office
C: Requires Dean approval or allocation
D: Requires EVC allocation based on Dean’s request
E: Requires impacted status (EVC approval) with CPB consultation and CAFA approval
F: Requires impacted status (EVC approval) with CPB consultation and CEP approval

1) Curricular efficiency (A/B): Undertaken by department. Actions include using same courses for major and non-major introductory programming for students without programming expertise, reduction in low-level service offerings, and increased class sizes. Motivated by the foreshadowing of this item, we included a streamlining of our BS requirements in the program catalog copy sent to CEP in November.

2) Practices in enforcing current qualification policy: Previously, exceptions were granted based on an interview. Last year we changed the policy so that exceptions now require written
appeals and are granted only in extraordinary cases. We have submitted updated catalog copy to CEP indicating that late declarations are an exception and thus require an appeal (which we intend to rarely grant).

3) **Promotion of UC Online courses appropriate to the major** (A): The courses we found are incompatible with our curriculum, using different languages and tools. See also the section on online education in the main document.

4) **Use of program’s discretionary resources** (A): Unclear what resources are suggested, the department has essentially no discretionary resources capable of providing significant relief. See also the response to 10).

5) **Consideration of alternate degree options (lower unit)** (B+D): We already have a lower-unit BA alternative to our BS degree.

6) **Refuse double majors after declaration deadline** (A): The number of double majors is relatively small, and lead to some of our better students and may improve diversity. Late double majors are approved by petition only.

7) **Establish 2-stage qualification policy** (B): In current SOE catalog copy submitted to CEP. See also the new Engineering major gating policy described in the main text.

8) **Priority enrollment for majors in some courses** (A): We are implementing this as fast as we are able. One problem is that our Registrar system does not per se allow priority enrollment according to major, so this is a process that needs to be done by registrar personnel by manually setting different deadlines for each major, a very labor intensive process that cannot be scaled to all classes. The Registrar can implement classes that are restricted to a set of majors only. We will work with Registrar to understand how to best use their resources. We may convert many classes to CS Majors only, and related majors when required by curriculum (e.g., CE, CM, TIM), except by instructor permission (permission codes).

9) **Restrict enrollment to majors in some courses** (B): This is more feasible than priority enrollment. We plan to restrict enrollment to the CS major and to majors with which we have MOUs (CE, CM, TIM, depending on the class),

10) **Additional resources for instruction from division** (C): The funding model is opaque to us, and additional resources might already be provided. However, the lack of capable lecturers greatly limits our ability to expand instruction. Long term resource commitments and recruiting help may be needed.

11) **Additional resources for instruction from the campus** (D): Temporary funds have been provided, but see 10 above.

12) **Qualification criteria that take capacity into account** (F): This is possible to pursue, even though it is not our preferred option. Rather than admit many students to the campus and then prevent them from declaring the CS major, it would be preferable to work on admissions criteria and quotas, so that the students who are not admitted at UCSC with CS major can pursue other options elsewhere.

13) **Higher admissions thresholds due to capacity** (E): This is precisely one of our goals, and it will become possible to pursue it once impaction status is established.
14) **Qualification capped to a specific number of students** (F) : See above for 12): we are waiting on impaction determination and suitable campus policies, but we would rather work on admissions than on qualification.

15) **Admissions capped to a specific target number of students** (E) : See above; this is one of our goals and it will become possible to pursue once impaction status is established.