1. GENERAL IMPERATIVES

1.1 Contribute to the betterment of the world.
As a field, computer science has given many entrepreneurs opportunities to become vastly successful. While many choose to make these successful men into role models, often their successes were a result of activities that reside in legal or ethical gray areas. Perhaps their work destabilized an entire industry, or profited off of other’s personal information, or provided no educational or social benefit to mankind. We should strive for success, not at the cost of the common good. Our success should be mutual, so that as we are raised up, so too is the world.

1.2 Respect and preserve the intrinsic rights of all persons.
The intrinsic rights of all individuals will not be subverted by other interests. Among these rights are the rights to freedom of conscience, speech, expression, and congregation; the rights against discrimination and abridgment of privacy; the rights against physical, mental, and emotional harm; and the rights to freely pursue opportunity and participate in society. Such a listing is not dispositive and does not exclude other liberties.

1.3 Promote diversity and inclusion on a global, local, and personal level.
Respect for the sacred values of equality, tolerance, and dignity transcends legally protected classes and demands a holistic, accepting approach to identity. To this end, computer scientists will fight manifestations of inequality and injustice however, whenever, and by whomever it is encountered.

1.4 Remember that the consumers, users, and other individuals affected by my work are human beings.
Given that the end of computer science is the betterment of the world, the computer scientist must not overlook the inherent worth of all persons. They must not work or act in intentional, reckless, or negligent disregard of that fundamental principle.

1.5 Give generous credit for intellectual property.
As all innovation rests upon the shoulders of many thinkers and designers before us and alongside us, credit must be afforded to those whose work contributes to an advancement. Beyond even the standards mandated by the law and expected by academia, computing professionals must be willing and eager to share recognition from a position of humility and grace.

2. AS A STUDENT

2.1 Seek full understanding of all code I write.
Plagiarism and copying grossly impede the furtherance of one’s abilities. Examples and sample code from any source are only as good as they are able to facilitate the
acquisition of knowledge. Computer science students will delve to whatever depths necessary to attain mastery over all written code.

2.2 Admit ignorance and seek to overcome rather than disguise inability. Computing is a craft that must be perfected through a continuous investment of time, work, and energy. Pride has no place in this pursuit, leading only to ineffectiveness and complacency. Students of computing must approach their studies with an open mind and an inquisitive nature.

2.3 Continue to study new technology and other advancements in the field, while deepening proficiency in past skills. Some have few interests and work to master a few fields, while others may be jacks-of-all-trades but not have remarkable proficiency in any individual area. In the old days of computer science, most would master a single language or discipline, such as Fortran or mainframes, and would become an indispensable member of the team as the expert of their given topic. Given how rapidly computer science is developing today, computer scientists cannot afford to limit themselves to a few single topics, as those topics may be replaced with something else in a few years. In order to keep up, we must keep ourselves open to new technology and advancements as we work with what we know.

2.4 Exhibit patience in helping or working with peers of all levels of knowledge and aptitude. Everyone who first starts learning in a given area begins with limited knowledge and skill, even those who one day become the masters of their field. We should always be patient when assisting peers who are less proficient than us, as we were once mentored ourselves. Our goal should be to teach others patiently, and strive to raise them to an equal or higher proficiency than our own so that we can continue to boost the standards of our field.

3. AS A TEAM MEMBER

3.1 Maximize code quality to make projects more stable and maintainable. In teams, each member writes their own elements of a large program, crafting a small mechanism of a larger machine. If one member’s fragment is faulty, then many problems will arise from both its individual operations and operations that interface with other pieces. In order to improve the work of the team and maximize the quality of the project as a whole, each member should seek to make their code as bug-free and stable as possible. This will also reduce the total maintenance that will need to be performed later on, which will limit the total work necessary for the team later on.
3.2 Write clear code and thorough documentation to ease others’ comprehension of project elements.
On a team, members will be constantly writing small, complex elements of large programs that will need to interface with others’ work, and vice versa. In order to facilitate communication amongst the team, members should write clean code and explain it in a way that is easy to understand. That way, the team will be able to work together much more efficiently, and future developers that may adapt or maintain that code in the future will be able to work with it more easily.

3.3 Justly distribute and perform collective responsibilities.
To each according to their abilities and availabilities, members of a team will allocate work fairly toward the collective interests of the team. No member being overburdened nor undervalued, each team member will be placed in a position to succeed.

3.4 Always look for opportunities to improve oneself and the team.
Members of a team should never be satisfied with their placement on the team, whether they be the worst or best member. As the worst member, they should seek to learn from their peers to improve their own skills, so they can meet the standards of the team and better serve the other members. Likewise, as the best member, they should seek to mentor other peers and raise others’ skill, and raise the standards of the entire team as a whole.

4. AS A PROFESSIONAL

4.1 Strive to discover fair compromises between public and corporate interests.
Often, the simplest solution that achieves a given corporation’s interests may go against the public’s interests. Although the given solution may be be legally acceptable, it may not be ethically acceptable for many. In order to provide the best outcome for all, a solution should be found that maximizes corporate interests while not infringing on public interests.

4.2 Present my qualifications honestly, accurately, and completely.
Understanding the need of companies to make effective employment decisions, the computing professional will strive neither to misguide nor to misinform throughout all stages of employment. Instead, the computing professional will do everything in their power to ensure that the employer and the employee may both make a well-informed decision.

4.3 Evaluate proactively the impacts and risks of my work.
The computing professional will design with care, not with haste. Deliberate
consideration will be taken to understand the ramifications of a product or segment of code before any adverse effects are unleashed unto the world.

4.4 Work to bring the best possible product to market.
   Even when competition is not in play, as a professional, one should always work to bring the best possible product to market. Shortcuts and substandard quality should be avoided--the product should meet or exceed a user’s expectations, and provide a beneficial impact.

4.5 Innovate on previous solutions, or establish brand new ones.
   Often, corporations will follow in the footsteps of others and put out similar or indistinguishable products in order to get a slice of the profits. Instead of remaining complacent with the status quo, one should always strive to raise the bar. Innovation should be a key goal, either by improving on others solutions in a valuable and meaningful way, or finding a brand new solution that outperforms the competition. Likewise, solutions should be sought even for problems most have not yet discovered.

4.6 Understand and respect the laws of all affected communities.
   Being cognizant of the cultural differences that pervade our global community, computing professionals will act from a place of acceptance and compliance. They will understand the profound weight of deference deserved by the law of a people and will only act in defiance of such a law when there can be no doubt that the law is directly opposed to and the act is directly aligned with the rights and dignities of the affected people.