Question 1 0 / 1 point

The author of the paper (David Parnas) describes some causes of software aging. Mark the alternatives that agree with the author's opinions on the causes of software aging.

Select 2 correct answer(s)

- [x] There are so many changes to a software system that make it impossible to software developers to change it in the future.
- [ ] The software system is based on an older operating system losing market share, which decreases its value to end users.
- [ ] When developers fail to release allocated memory, software systems slow down and age.
- [x] Software system's files may grow and require pruning, with swap and file space degrading performance.
- [ ] Software developers do not change software according to users' expectations, who view the the software as old and outdated.
- [ ] Inappropriate changes are made to the software system without abiding by the original design decisions.

Question 2 0.667 / 1 point

The author of the paper (David Parnas) describes some costs associated to software aging. Mark the alternatives that agree with the author's opinions on the costs of software aging.

Select 3 correct answer(s)

- [ ] A large code base eventually leads software developers to not choose new technologies, preventing system upgrades that will adapt the software system to improved technologies.
- [x] Deteriorating software structure usually degrades space/time performance of the software system.
- [ ] Changes made to aging software tend to introduce more bugs.
- [ ] Restructuring the software system becomes impossible because of previous bad design decisions.
- [ ] Adding more developers to the software project only brings more communication challenges, making it difficult to perform software changes.
- [x] Software documentation costs becomes too high because documentation was left to be done after software release.
- [x] Developers find it increasingly hard to modify the system to keep up with the market.

Question 3 1 / 1 point

The author of the paper (David Parnas) describes some preventive measures to avoid or postpone software aging. Mark the alternatives that agree with the author's opinions on the preventive measures against software aging.

Select 3 correct answer(s)

- [ ] Use configuration management tools to record all the commits by developers, which will assure knowledge of design decisions.
- [ ] Stop the software deterioration by removing a large part of the software system that is causing problems.
- [x] Use reviews from other professionals, since second opinions are important in every profession, software professionals included.
- [x] Design the software system for changes, by using information hiding and abstraction techniques.
- [ ] Use new technologies that incorporate design decisions in the source code.
- [x] Improve documentation by recording design principles and decisions in an useful form to future maintainers.
- [ ] Accept the fact that software systems will age, and prepare for retiring the aging software system.

Question 4 0.75 / 1 point

The author of the paper (David Parnas) describes some corrective measures to treat software aging. Mark the alternatives that agree with the author's opinions on the corrective measures to treat software aging.

Select 4 correct answer(s)

- [ ] Slow down software deterioration by reintroducing structure in the software system.
- [ ] Use new technologies that incorporate design decisions in the source code.
- [x] Upgrade the quality of the documentation retroactively.
- [ ] Use configuration management tools to record all the commits by developers, which will lead to better knowledge of design decisions.
- [x] Incrementally modularize the software system, by using appropriate information hiding and abstraction choices.
- [x] Reduce the software deterioration by removing a large part of the software system that is causing problems.
- [ ] Invest in educating the software developers to be aware of the software aging process.
- [x] Use software metrics analysis to better understand the parts of the software system that are aging.

Question 5 0.5 / 1 point

The author of the paper (David Parnas) describes some necessary changes to software organizations to better deal with software aging. Mark the alternatives that agree with the author's opinions on the those planning ahead changes.

Select 4 correct answer(s)

- [ ] Design must be created and documented during coding, not after it is done.
- [x] Designing for change should be an integral part of software organizations.
- [ ] Financial planning is also needed for replacement of aged software.
- [x] Today's pressures should not prevent better standards on structure and documentation.
- [ ] Measures against software aging need to be taken as part of designer-developer conversations in the software team.
- [x] Software developers need to learn to use software structure analysis tools to better deal with aging software.
- [ ] Invest in educating the software developers to be aware of the software aging process.
- [x] There must be more upfront design instead of rushing to code solutions as in current agile development processes.

Question 6 1 / 1 point

The author of the paper (David Parnas) describes some barriers to progress in the software engineering profession. Mark the alternatives that agree with the author's opinions on the those barriers to progress.

Select 4 correct answer(s)

- [ ] There is a misunderstanding between software evolution and software maintenance goals, which leads to poor maintenance practices.
- [x] There is a lack of professional identity and education on software engineering as a profession.
- [ ] Incremental and iterative processes avoid appropriate design and documentation practices.
- [ ] Software developers generally do not know appropriate reverse engineering and reengineering practices.
- [x] There is intellectual isolation from different industries that rely on software.
- [x] Software engineering researchers generally write papers focused on other researchers, not on software practitioners.
- [x] Software industry is focused on short-term changes.