CS6235 Quz 1 Exam: Mar 10 2021

Maximum marks = 22, Time: 1.00 hr

Name: ____________________ Roll: __________

- Write your roll number on every sheet of the answer book.
- Each of the MCQ/MSQ question is for five marks. Marks will be given based on the answer(s) and the working shown in the paper. If the working is not shown, no marks will be awarded.
- **Negative marking**. Each incorrect True/False answer will lead to a deduction of 0.5 mark.
- Advise: work out each question separately and legibly.
- Total marks=25.

---

1. Consider the following snippet of (partly) Java style code.

```
class item {
 lock m;
 job j;
}

main(List<item> L){
 if (L.isEmpty()) return;
 while  (!(every item.j in L is set to null)) {
  y = pickOneItemNotYetDone(L);
  if (some-condition) {
   create-thread-to-execute clearWork(y);
  } else {
   create-thread-to-execute processWork(y);
  }
 }
 wait-for-all-threads-to-terminate();
}
```

```
void processWork (item x){
 acquire (x.m); // blocking

 if (x.j == null) {release (x.m); return; }

 release (x.m);
 // Do the job x.j - code not shown.
 acquire (x.m);
 x.j.done = true;
 release (x.m);
 clearWork(x);
}
void clearWork (item x){
 acquire(x.m);
 if (x.j != null) x.j = null;
 release(x.m);
}
```

Which of the following option(s) is/are correct?

(a) Code has no data-race

(b) Code may go into a deadlock

(c) Code may throw NullPointerException

(d) Code may get into an infinite loop

2. Consider the following snippet of code:

```
S // serial part. Time taken = Ts.
P // Can be parallelized. Time taken = Tp
```

Assume γ = fraction of time taken by the serial part of the code. If `P` is a parallel for-loop that creates at most 2 threads, whose bodies cannot be further parallelized, then the maximum speed up using N (for large values of N) cores =

(a) $\frac{2}{1+\gamma}$

(b) $\frac{2}{1-\gamma}$

(c) $\frac{N}{1+\gamma}$

(d) $\frac{1}{\gamma}$

3. Consider the following piece of code:

```
S1: AtomicInteger x = 1
parallel-for-loop (i=0;i<n;++i){ // creates a n parallel threads
  S2: A[i] = i;
  S3: B[i] = x.incrementAndGet();
}
// wait for all the iterations to get over.
S4:
```

If X $\leftarrow$ Y, indicates that X happens-before Y, then which one of the following options is true:

(a) S1 $\leftarrow$ S2, S1 $\leftarrow$ S3, S2 $\leftarrow$ S4, S3 $\leftarrow$ S4

(b) S1 $\leftarrow$ S2, S2 $\leftarrow$ S3, S3 $\leftarrow$ S4

(c) S1 $\leftarrow$ S2, S2 $\leftarrow$ S3, S3 $\leftarrow$ S2, S2 $\leftarrow$ S4, S3 $\leftarrow$ S4

(d) None of the above

4. **Undeclared and Uninitialized variables**:
In a new language a variable's use is considered *undefined*, if it is both undeclared and uninitialized. Assume that only the following types of statements are present: i) variable declaration (for example, int x), ii) copy statement (for example, x = constant, or x = y), iii) if-statement, iv) loops.
We want to use the iterative worklist based algorithm (studied in the class to) identify undefined uses of variables. Specify the lattice you want to use, and how to write the transfer functions for each of the four possible statements. Also mention how you will use your analysis to report the errors. Assume each statement to be an independent node.

Section 2. True/False (1 mark each)

________ For the code shown in Q2, Amdahl's law is not applicable if `Tp` = 0.

________ For the code shown in Q2, speed up obtained on N cores cannot be computed, if `Ts` = 0.

________ Say a task A1 is created by a thread T0. Say A1 creates a thread T1. A1 and T1 can share the heap, but not the stack.

________ The number of elements in the constant-propagation lattice is bound by the program size (N), Or in other words $O(N)$.

________ We can write a function where the number of nodes in the CFG is greater than the number of statements in the function.