COMP2303/7306 2011
Mid-semester exam
Open Book, no commuication devices, approved calculators only
Time allowed: 35 minutes
Reading time: 5 minutes

Do not use this paper for study without reading the warning on the course website.

Instructions:

- In all questions circle the most correct answer.
- In questions where knowledge of the computer's hardware is required, assume the following:
  - A 32bit address space.
  - 4KB pages with 4 byte page table entries.
  - Uses two level page tables.
  - L1 and L2 cache are considered part of memory.

---

1) Which lines in the following would cause warnings/errors under `gcc -c -ansi -pedantic -Wall`?

```
int f(char c) {                                          [1]
    if ((c<'a') || (c>'z')){                             [2]
         return;                                         [3]
    }                                                    [4]
    char i;                                              [5]
    for (i=c;i<='z';++i){                                [6]
       printf("%c"); // print remaining                  [7]
    }                                                    [8]
    return 1;                                            [9]
}                                                        [10]
```

(a) 1
(b) 2
(c) 3
(d) 5
(e) 6
(f) 7
(g) 9
(h) 3, 5
(i) 6, 7
(j) 3, 5, 7
(k) 2, 3, 5
(l) 2, 5, 7
(m) 2, 6, 9
(n) 3, 5, 9
(o) 1, 2, 3, 5
(p) none of the above

---

2) Which of the following `svn` commands do not require a commit?

(a) add

(b) remove

(c) status

(d) update

(e) diff

(f) add, diff

(g) update, remove

(h) status, diff, update

(i) add, remove, update, diff, status

(j) We can't be certain/none of the above.

---

3) Consider the following program:

```
#include <stdio.h>
int main(int argc, char** argv) {
    int i;
    for (i=0;i<3;++i) {
        fprintf(stderr, "%d is ",i);
        if (i%2) {
           fprintf(stderr, "even\n");
        } else {
           fprintf(stderr, "odd\n");
        }
    }
    return 0;
}
```

The program is compiled and run with the following command:

```
./a.out | grep -v even
```

Which lines will appear on the console?

(a) nothing will appear

(b)
```
0 is even
1 is odd
2 is even
```

(c)
```
0 is even
2 is even
```

(d)
```
1 is odd
```

(e) The shell command is invalid

(f) none of the above.

4) We wish to find the 7th line in a file called `data` which contains the word "fish". Which of the following lines would do this?

(a) `grep data fish | head -7 | tail -1`

(b) `grep fish data | tail -7 | head -1`

(c) `cat data | head -7 | tail -1 | grep fish`

(d) `cat data | grep fish | head -7 | tail -1`

(e) `grep fish | head -7 | tail -1`

(f) `head -7 | cat data | tail -1 | grep fish`

(g) none of the above.

---

5) After executing the following code fragment what are the values of w, x, y, z.

```
int w=0;
int x=6;
int y=3;
int z=2;
if (x=y==z){
    z=4;
} else {
    z=9;
}
if (w);{
    w=7;
}
```

(a) w=7; x=0; y=2; z=4;

(b) w=0; x=6; y=2; z=4;

(c) w=7; x=0; y=3; z=9;

(d) w=0; x=0; y=3; z=4;

(e) w=7; x=6; y=2; z=9;

(f) It is not legal C.

(g) Segmentation fault.

(h) We can't be certain/none of the above.

6) What are the values of i and j after executing the following code.

```
int i=4;
int j=2;
switch (i++-j) {
    case 3:i++; break;
    case 1:j++; break;
    case 2:j+=2;
    case 5:i+=2;
    default:i+=5;
}
```

(a) i=5; j=2;

(b) i=6; j=2;

(c) i=7; j=4

(d) i=9; j=2;

(e) i=11; j=2;

(f) i=12; j=4;

(g) It is not legal C.

(h) Segmentation fault.

(i) We can't be certain/none of the above.

---

7) What type are `var, foo, baz` in the following?

```
long* var, foo[7], baz;
```

(a) var - pointer to long, foo - array of 7 pointers to long, baz - pointer to long

(b) var - pointer to long, foo - array of 7 longs, baz - long

(c) var - long, foo - array of 7 pointers to long, baz - long

(d) var - pointer to long, foo - array of 7 longs, baz - pointer to long

(e) var - long, foo - array of 7 longs, baz - long

(f) It is not legal C.

(g) We can't be certain/none of the above.

---

8) Which of the following declares `foo` to be:
a function which returns a string and takes a pointer to a function which takes a char and returns a char.

(a) char* (*foo)(char (*)(char))

(b) char *foo(char(*)(char))

(c) char (*foo)((*)())

(d) char* (*foo)(char (*)(char))

(e) It is not possible to write such a declaration.

(f) We can't be certain/none of the above.

---

9) What is the result of the following code?

```
int x=2, y=3, z=5;
int* py=&x;
int* px=&y;
*px=x+y;
z=*px;
x=*px-*py;
*px=y-*py;
```

(a) x=0, y=3, z=5

(b) x=2, y=3, z=5

(c) x=3, y=2, z=5

(d) x=2, y=3, z=2

(e) x=2, y=3, z=3

(f) It is not legal C.

(g) Segmentation fault;

(h) We can't be certain/none of the above.

.

---

10) What is the result of the following code?

```
int x=0, y=2, z=4;
int *px, *py, *pz;
py=&z;
pz=&y;
x=x+4;
*px+=4;
```

(a) x=0, y=2, z=4

(b) x=4, y=2, z=4

(c) x=8, y=2, z=4

(d) x=4, y=4, z=4

(e) It is not legal C.

(f) Segmentation fault;

(g) We can't be certain/none of the above.

---

11) Consider the following function:

```
void f(int x) {
    x--;
    fprintf(stderr, "B");
    if (fork()) {
        fprintf(stderr, "C");
        if (x>1) {
            f(x);
        }
    } else {
        fprintf(stderr, "A");
    }
    fprintf(stderr, "B");
}
```

When f(3) is called, how many B's will be output?

(a) 2

(b) 3

(c) 4

(d) 5

(e) 6

(f) 7

(g) 8

(h) 9

(i) none of the above.

---

12) When is a zombie process **not** created?

(a) When a segmentation fault occurs.

(b) When _exit() is called.

(c) When the `main()` function returns.

(d) When its parent was `wait()`ing for it.

(e) none of the above.

---

13) The CPU access the following (base 10) virtual addresses in sequence.
45055, 45056, 8392710, 45090, 8392714, 45091
How many accesses to memory are required?

(a) 4

(b) 5

(c) 6

(d) 9

(e) 10

(f) 11

(g) 12

(h) 15

(i) 16

(j) none of the above.

---

14) What is the largest amount of memory the page table for a single process can occupy?

(a) 3072KB

(b) 4096KB.

(c) 4100KB.

(d) 1048576KB.

(e) 4GB.

(f) none of the above.

15) Given the following page table, map the (base 10) virtual addresses to physical addresses.

0	200
1	201
2	22
3	invalid
...	invalid
25	24
...	invalid

Addresses: 500, 8200, 102400

(a) 500, 8200, 102400

(b) 819700, 90120, 98304

(c) 823796, 90120, 102400

(d) 819700, 823304, 102400

(e) 823796, 823304, 98304

(f) A page fault will occur

(g) A segmentation fault will occur

(h) none of the above