UIUC CS241 Practice Quiz 2 (40 min)

1. (1 point) Which one of the following is true for `fork()` ?

(A) Creates a new process by reloading the program - the child starts at `main()`
(B) Creates a new file handle for standard int
(C) Installs a new fork handler
(D) Creates a new file handle for standard out
(E) Creates a new process by cloning the existing process; the child starts by `fork()` returning 0

2. (1 point) When will `fork()` return -1 ?

(A) When a child needs to be restarted
(B) When the parent is the first process
(C) In the child process
(D) In the parent process
(E) If `fork` failed

3. (1 point) Which one of the following is NOT true?

(A) The default action of `SIGCHLD` is to do nothing
(B) Pressing `CTRL-C` will send a `SIGINT` signal to the process
(C) The default action of `SIGALRM` is to quit the process
(D) Signals are software interrupts - they are handled concurrently by signal handlers.
(E) Signals can be sent to other processes using the 'signal' function

4. (1 point) Which one of the following changes the process's current directory to the user's home directory?

(A) `pwd( environ[0] )`
(B) `chdir( getenv("HOME") )`
(C) None of the other responses are correct
(D) `chdir( environ[ getenv("HOME")] )`
(E) `pwd( environ[UHOME] )`

5. (1 point) If `malloc` fails (returns NULL) will the following program crash (seg fault)? If so, where?

```
1 void * ptr1 = (void*) malloc(16);
2 int ** ptr2 = (int**) ptr1;
3 int *** ptr3 = & ptr2;
4 void* ptr4= (void*) &ptr1;
```

(A) Line 1
(B) Line 3
(C) Line 4
(D) Line 2
(E) None of the other responses are correct

6. (1 point) Which response best describes the common system programming pattern to run another program and wait for it to finish?

(A) Child process execs then child forks and child waits
(B) Parent process forks then parent execs and child waits
(C) Parent process forks then child execs and parent waits
(D) Parent process execs then child forks and child waits
(E) Child process waits then parent exec and child forks

7. (1 point) Which one of the following best describes the code below?

```
int *p = (int *)malloc(sizeof(int));
p = NULL;
free(p);
```

(A) There are no errors in the code
(B) Memory leak
(C) The program may crash as `free()` is called on a NULL pointer
(D) Dangling pointer
(E) Compiler error: `free` cant be applied on NULL pointer

8. (1 point) What will be the most likely last thing printed by the following program?

```
int main() {
   int c = fork();
   printf("c=%d : pid=%d ppid=%d\n",c, getpid(),getppid() );
   if(c>0) return 97;
   sleep(4);
   printf("Answer: %d\n",getppid());
   return 80;
 }
OUTPUT:
c=0 : pid=97 ppid=90
c=97 : pid=90 ppid=80
---- ?
```

(A) `Answer:  1`
(B) `Answer:  80`
(C) `Answer:  97`
(D) `Answer:  90`
(E) None of the other responses are correct

9. (1 point) Which one of the following is NOT true for a child process created by `fork`?

(A) Gets its own complete copy of the parent's process memory
(B) Starts after the parent process has finished
(C) Inherits signal handlers defined using `signal`
(D) Inherits (shares) the parent's open file streams

10. (1 point) Which one of the following is NOT true?

(A) `char** environ` should be declared `extern`
(B) `argv[1]` is the first argument because `argv[0]` is the program name
(C) `static` variables are automatic because they stored in the stack memory.
(D) `malloc` allocates memory on the heap
(E) The last entry of string arrays `argv` and `environ` is always `NULL`

11. (1 point) `puts(ptr)` is equivalent to

(A) `scanf("%s\n",ptr)`
(B) `signal(SIGINT,ptr)`
(C) `printf("%s\n",ptr)`
(D) `fprintf(stderr,"%s\n",ptr)`
(E) `ptr=getchar()`

12. (1 point) Which one of the following is most likely to print `Hi` only once?

(A) `execl("nosuchfile","nss",(char*)NULL); write(1,"Hi",2);`

(B) `execl("/bin/ls","ls",(char*)NULL); write(1,"Hi",2);`

(C) `puts("Hi");if(1) puts("Hi");`

(D) `fork(); write(1,"Hi",2);`

13. (1 point) How many times will `!` be printed when this program is run in a terminal?

```
  int main() {
     printf("!");
     fork();
     fork();
     exit(0);
     return 0;
   }
```

(A) 1

(B) 5 or more

(C) 2

(D) 4

(E) 3

14. (1 point) Which one of the following best describes the correct line 4 to read the line and store the result in the buffer and score variables?

```
1  char* buffer = (char*) malloc(16);
2  int score,res;
3  char* line = "Pointers 123";
4  _______  ?
```

(A) `res = sscanf(line, "%15s %d",buffer, score);`

(B) `res = sscanf(line, "%15s %d", *buffer, *score);`

(C) `res = sscanf(line, "%15s %d", &buffer, &score);`

(D) `res = sscanf(line, "%15s %d",buffer, &score);`

(E) `res = sscanf(line, "%15s %d", &buffer, score);`

15. (1 point) Which one of the following is the best description of POSIX process control? When a child process finishes (or temporarily stops) ...

(A) The init (process 1) is sent a `SIGUSR1` signal

(B) The parent process is sent a `SIGCHLD` signal

(C) The child process is re-assigned a new parent process

(D) All siblings are notified with a `SIGQUIT` signal

(E) The process is automatically restarted

16. (1 point) Which one of the following is true for `gets`?

(A) Returns an integer pointer

(B) The function `gets` is the recommended function to read lines of input into a buffer

(C) Can only read text data from a file

(D) Allows a buffer overflow if the input line is longer than the buffer

(E) Can only read binary data from a file

17. (1 point) Which one of the following is the best choice of missing code? Choose the correct snippet so that the program uses `string.h` functions to will display the message when the program is started with `"-h"` option

```
int main(int argc, char*argv[]) {
  // If no arguments or just -h, then show a message and quit:
  if(argc ==1 ||   _______    ) help_message_and_quit();
```

(A) `argv[1] == "-h"`

(B) `argv[0] == "-h"`

(C) `0==strcmp(argv[1],"-h")`

(D) `0==strcmp(argv[0],"-h")`

(E) `0==streq(argv[0],"-h")`

18. (1 point) Four students were asked to write four alternative ways to print `Hello World!` to standard output. Carefully read the four functions below and for each one, decide if it will print `Hello World!` without error. Choose the most accurate response below.

```
void A() { char *s=(char*)malloc(100); strcpy(s,"Hello World!\n"); puts(s); free(s); }
void B() { char s[100]; *s=0; strcat(s, "Hello World!\n"); puts(s); }
void C() { static char s[100]; sprintf(s,"Hello "); strcat(s,"World!\n"); write(1,s,strlen(s)); }
void D() { char *s = "Hello "; strcat(s, "World!"); printf("%s\n", s); }
```

(A) All 4 functions are correct

(B) None of the functions are correct

(C) Only 1 function is correct

(D) 3 functions are correct

(E) 2 functions are correct

19. (1 point) Which of the following is NOT true for `getline`?

(A) `getline` returns the number of characters read (possibly including a newline character at the end)

(B) Is used to convert a character array into integer and floating point values

(C) It's important to set both capacity to zero and the character pointer to `NULL` before the first call to `getline`

(D) `getline` arguments include a pointer to an int and a pointer to a pointer to char, so it can modify their contents.

(E) To avoid a memory leak, call free on the buffer after the last call to `getline`

20. (1 point) Which response best describes the following buggy code that, when executed on a 32 bit machine (pointers require 4 bytes), is suppose to create a 16x16 2D character array?

```
1  char ** array;
2  array = (char**) malloc(16 * sizeof(char)); // bug here
3  if( ! array ) return 1; // malloc failed
4  int i = 0;
5  for(; i < 16;i++)
6    array[i] = malloc(16 * sizeof(16));
```

(A) Insufficient memory allocated (line 2) causes a buffer overflow when i is 16 and higher

(B) Insufficient memory allocated (line 2) causes a buffer overflow when i is 8 and higher

(C) Insufficient memory allocated (line 2) causes a buffer overflow when i is 0

(D) Insufficient memory allocated (line 2) causes a buffer overflow when i is 4 and higher

(E) Insufficient memory allocated (line 2) causes a buffer overflow when i is 12 and higher

21. (1 point) Which one of the following is NOT true?

(A) `alarm(5)` will send an asynchronous signal `SIGALRM` to the process in 5 seconds

(B) If `open` succeeds it will return the smallest unused non-negative integer

(C) The default SIGALRM handler does nothing

(D) Dead processes (zombies) still take up space in the system's process table. If the system table is full no new processes can be created.

(E) Standard error stream (stderr, file descriptor 2) is not buffered

22. (1 point) Which one of the following is NOT true for the following line?

```
waitpid(id,&status,0); // Assume waitpid is successful
```

(A) Is used to cleanup zombies from the kernel's process table

(B) The `status` variable can be queried with macros `WIFEXITED,WIFSIGNALED,WEXITSTATUS` to extract the status of the child process

(C) Is used to prevent child processes from starting immediately

(D) If the child exited normally then `status` it includes the lowest 8 bits of the child's exit value

(E) The status variable is an `int; &status` means the address of the integer

23. (1 point) Which one of the following prints `H` to the standard output stream?

```
char* ptr = "H";
________?
```

(A) `printf("%p",ptr);`

(B) `write(1,ptr,strlen(ptr));`

(C) `fprintf(stderr,"%s",ptr);`

(D) `puts(* ptr);`

(E) `write(sizeof(ptr), ptr, stdout);`

24. (1 point) A zombie is created when

(A) A parent sends the child process a `SIGZOMB` signal

(B) A child calls `exec`

(C) A child doesn't wait on a parent

(D) The parent finishes before the child process

(E) A parent doesn't wait on a finished child