CSC 357 – Name: _____

CAL POLY
SAN LUIS OBISPO

CSC 357: Systems Programming

Midterm

- Put your name on all sheets.
- No books, notes or other type of written material are allowed.
- Mobile phones are not allowed.
- Show your work neatly for partial credit.
- You must show your work to receive any credit.
- Max points: 100 + 4 bonus.
- You can use the backside of the sheets.
- Do not get distracted by answering unnecessary questions.
- Read the statement below and sign your name.
- Use your laptop. You are not allowed to use anything else than your IDE except: http://man7.org/linux/man-pages/man2/mmap.2.html and https://linux.die.net/man/2/munmap

I affirm that I neither will give nor receive unauthorized assistance on this examination. All the work that appears on the following pages is entirely my own.

Name:

_____

Signature:

_____

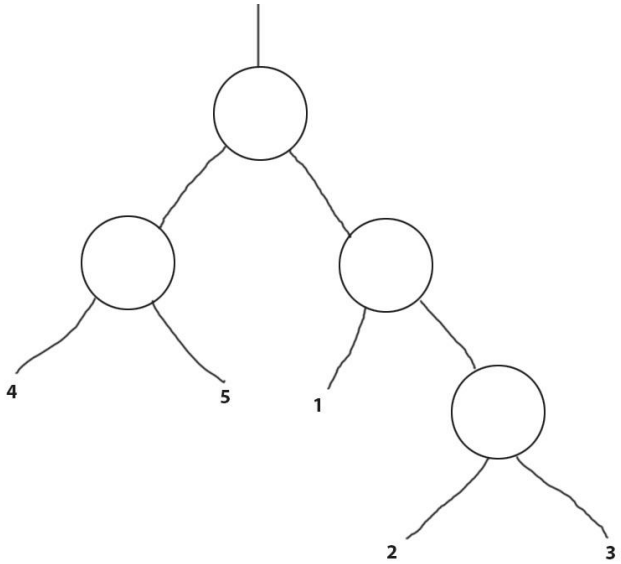
Total points	Letter grade

CSC 357 – Name: _____

Task 1: forking (40 pts)

Points

Print the numbers in the order seen on the image. Use fork() and wait() to start new processes in 100% accordance to the image. You need to decide where child- and parent processes are. (Sorry for the bad line drawing, they will be better on the real midterm ;)



Task 2: lists (40 pts)

Points

Program a double linked list from scratch. Start with an empty list head, which should be global. The data segment is an integer. Implement following functions:

new_element() //should read an integer, malloc a new element and puts it at the front (!) of the list
printlist()//prints the whole list (data segment and addresses prev & next)
remove_element(int d)//finds all elements with data "d" and removes them from the list (free).
delete_list()// deletes (free) the whole list)
Use following structure:

```
typedef struct listelement
{
struct listelement *prev,*next;
int number;
}listelement;
```

CSC 357 – Name: _____

Task 3: read from bitmap

Points

Open a bitmapfile (the two fileheader/fileinfoheader structures will be provided) and save all colors as their negative (e.g. color = 255 – color;)

Task 4: read from binary file

Points

Open a binary file “file.bin”. The content of the file is following structure:

```
typedef struct datafile
{
char type;
int size;
char type2;
int offset;
char text[10];
}datafile;
```

Make an instance of datafile in your main function. Read from the file into this structure (its up to you how). Print all elements of “datafile”. Don’t forget to close the file pointer.

Task 5: memory allocation

Points

Write a program, that reads text from the keyboard. One word is fine, no need for whole sentences. Allocate exactly what you need for each text on the heap with mmap().
If you enter the same text again, remove this word from the heap with munmap().
If you enter “print”, write all words entered so far.
Maximum word count is 10, so you don’t have to (but you can) use a list. An array like: char *text[10]; will do too. Be careful not to print deleted words again!

CSC 357 – Name: _____

Task 6: multi-tasking

Points

Write a program, that allocates 1000 integers on the heap. Now fork twice (!) and use all four processes to write the index of the array into its content (arr[i] = i). Every process should write into their own quarter. The order doesn't matter. Wait until all processes are ready, then print the whole array.

Task 7: answer the 20 questions on PolyLearn

Points

The 20 questions will be from the pool of all lecture quizzes so far plus the Monday questions before the midterm.