

# yifany2

---

## 1. Write a program that uses write() to print out "Hi! My name is <Your Name>".

```
int main() {
    write(1,"Hi! My name is Yifan Yang",25);
    return 0;
}
```

Grade: 100%

## 2. Write a function to print out a triangle of height n to standard error.

```
void write_triangle(int n) {
    if (n == 0) return;
    write_triangle(n-1);
    int i = 0;
    for (i = 0; i < n; i++) {
        write(STDERR_FILENO, ".",1);
    }
    write(STDERR_FILENO, "\n",1);
}
```

Grade: 100%

## 3. Take your program from "Hello, World!" and modify it write to a file called "hello\_world.txt".

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
/*void write_triangle(int n) {
    if (n == 0) return;
    write_triangle(n-1);
    int i = 0;
    for (i = 0; i < n; i++) {
        write(STDERR_FILENO, ".",1);
    }
    write(STDERR_FILENO, "\n",1);
}*/

int main() {
    // write(1,"Hi! My name is Yifan Yang",25);
    // write_triangle(5);
    mode_t mode = S_IRUSR | S_IWUSR;
    int fildes = open("hello_world.txt", O_CREAT | O_TRUNC | O_RDWR, mode);
    write(fildes,"Hello, World!",13);
    close(fildes);

    return 0;
}
```

Grade: 100%

**4. Take your program from "Writing to files" and replace write() with printf().**

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <stdio.h>

int main() {

    mode_t mode = S_IRUSR | S_IWUSR;
    close(1);
    int file = open("hello_world.txt", O_CREAT | O_TRUNC | O_RDWR, mode);

    printf("hello_world\n");

    close(file);

    return 0;
}
```

Grade: 100%

**5. What are some differences between write() and printf()?**

write is a system call while printf is a library function that wrap up the information in a buffer and send it to stdout.

Grade: 100%

**6. How many bits are there in a byte?**

at least 8 bits

Grade: 100%

**7. How many bytes are there in a char?**

1

Grade: 100%

**8. How many bytes are each of the following on your machine?**

4, 8, 4, 8, 8

Grade: 100%

**9. Refer to code snippet below. If the address of data is 0x7fbd9d40, then what is the address of data+2?**

0x7fbd9d50

Grade: 100%

**10. What is data[3] equivalent to in C?**

`*(data+3)`

Grade: **100%**

**11. Why does the code snippet below segfault?**

`"hello" only can be read.`

Grade: **100%**

**12. What does sizeof("Hello\0World") return?**

`12`

Grade: **100%**

**13. What does strlen("Hello\0World") return?**

`5`

Grade: **100%**

**14. Give an example of X such that sizeof(X) is 3.**

`char* X = "ab";`

Grade: **100%**

**15. Give an example of Y such at sizeof(Y) might be 4 or 8 depending on the machine.**

`long Y;`

Grade: **100%**

**16. What are two ways to find the length of argv?**

The length of argv is stored in argc.  
OR write a loop to count.  
`int count = 0;  
while (argv[count])  
{  
count++;  
}`

Grade: **100%**

**17. What does argv[0] represent?**

The execution of the program.

Grade: **100%**

**18. Where are the pointers to environment variables stored?**

It is stored in memory using by the shell that we don't need to declare them in the program

Grade: **100%**

**19. Refer to the code snippet below. What are the values of sizeof(ptr) and sizeof(array)?**

4, 6

Grade: **75%**

**Feedback:** The question says the size of pointers is 8 bytes.

**20. What data structure manages the lifetime of automatic variables?**

stack

Grade: **100%**

**21. If I want to use data after the lifetime of the function it was created in ends, where should I put it? How do I put it there?**

in the heap. malloc some memory and get a pointer to it.

Grade: **100%**

**22. Fill in the blank: "In a good C program, for every malloc, there is a \_\_\_\_".**

free

Grade: **100%**

**23. What is one reason malloc can fail?**

We've used up all heap memory.

Grade: **100%**

**24. What are some differences between time() and ctime()?**

time() returns a value of time\_t, while ctime takes in a time\_t and returns a pointer of char.

Grade: **100%**

**25. What is wrong with this code snippet?**

It free the same pointer twice.

Grade: **100.0%**

## 26. What is wrong with this code snippet?.1

It uses a memory that has been freed.

Grade: 100%

## 27. How can one avoid the previous two mistakes?

set the pointer to be NULL.

Grade: 100%

## 28. Create a struct that represents a Person, and then make a typedef, so that "struct Person" can be replaced with a single word.

```
struct Person{
    char* name;
    int age;
    struct Person** list;
};
typedef struct Person person_t;
```

Grade: 100%

## 29. Now make two persons on the heap, "Agent Smith" and "Sonny Moore", who are 128 and 256 years old respectively and are friends with each other.

```
person_t* p1 = (person_t*) malloc(sizeof(person_t));
person_t* p2 = (person_t*) malloc(sizeof(person_t));
p1->name = (char*) malloc(sizeof(char)*50);
p2->name = (char*) malloc(sizeof(char)*50);
strcpy(p1->name,"Agent Smith");
strcpy(p2->name,"Sonny Moore");
p1->age = 128;
p1->list = malloc(sizeof(person_t)*10);
p1->list[0] = p2;
p2->list = malloc(sizeof(person_t)*10);
p2->list[0] = p1;
p2->age = 256;
```

Grade: 100%

## 30. Create functions to create and destroy a Person on the heap.

```
person_t* create(char* str, int a) {
    person_t* res = (person_t*) malloc(sizeof(person_t));
    res->name = malloc(sizeof(char)*50);
    strcpy(res->name,str);
    res->list = malloc(sizeof(person_t)*10);
    int i;
    for (i = 0; i < 10; i++) {
        res->list[i] = NULL;
    }
}
```

```

    }
    res->age = a;
    return res;
}
void destroy(person_t* p) {
    free(p->name);
    free(p->list);
    int i = 0;
    for(; i < 10; i++) {
        if(p->list[i] != NULL)
            p->list[i] = NULL;
    }
    free(p);
}

```

Grade: 100%

**31. What functions can be used for getting characters from stdin and writing them to stdout?**

getchar(), putchar()

Grade: 100%

**32. Name one issue with gets().**

When you input a string that has more chars than the buffer can hold, it will affect the buffer of printf when printing to stdout.

Grade: 100%

**33. Write code that parses the string "Hello 5 World" and initializes 3 variables to "Hello", 5, and "World".**

```

int main() {
    char* data = "Hello 5 World";
    char a[20];
    char b[20];
    int num;
    sscanf(data,"%s %d %s",a,&num,b);
    //puts(a);
    //puts(b);
    //printf("%d\n",num);
}

```

Grade: 100%

**34. What does one need to define before including getline()?**

#define \_GNU\_SOURCE

Grade: 100%

**35. Write a C program to print out the contents of a file line-by-line using getline().**

```
#define _GNU_SOURCE

#include <stdio.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
#include <stdlib.h>

int main() {
    char* buffer = NULL;
    size_t capacity = 0;
    ssize_t numchars = 0;
    FILE* file;
    char* file_name;
    scanf("%s",file_name);
    file = fopen(file_name,"r");
    if (file == NULL) return 0;
    else {
        while ( numchars = getline(&buffer,&numchars, file) != EOF)
            puts(buffer);
    }
    fclose(file);
    free(buffer);

    return 0;
}
```

Grade: **100%**

**36. What compiler flag is used to generate a debug build?**

-g

Grade: **100%**

**37. You modify the makefile to generate debug builds and type make again. Explain why this is insufficient to generate a new build.**

You need type make debug instead of make

Grade: **0%**

**38. Are tabs or spaces used to indent the commands after the rule in a Makefile?**

They are essential. Without tab, there will be a syntax problem.

Grade: **100%**

**39. What are the differences between heap and stack memory?**

Stack memory will be automatically freed, but heap memory should be freed manually or it will leave there.

Grade: **100%**

**40. Are there other kinds of memory in a process?**

A data segment and a text segment.

Grade: **100%**

**41. Convert your a song lyrics into System Programming and C code covered in this wiki book and share on Piazza.**

[No response]

**42. Find, in your opinion, the best and worst C code on the web and post the link to Piazza.**

[No response]

**43. Write a short C program with a deliberate subtle C bug and post it on Piazza to see if others can spot your bug.**

[No response]

**Final grade: 96.875%**