

**Problem 1 (20 points)-True/False** Submit a document called Answers.doc

---

Answer the following true/false questions. You must correctly state **WHY** your answer is true or false in order to receive credit.

```
#include <stdio.h>
#define SIZE 4

int foo(int *a)
{
    return *a+4;
}

int foo1(char * w[])
{
    int i;
    for(i=0;i<SIZE;i++)
    {
        printf("%s ", w[i]);
    }

    return i;
}

main ()
{

    int continue1=8;
    int num, num2;
    int *ptr=&num;

    char * words[]{"one", "two", "three", "four"};

    while(continue1)
    {
        printf("\n\nEnter a number: 1, 2 or 3 to quit: \n");
        scanf("%d", ptr);

        if(num==1)
        {
            printf("Enter a number: ");
            scanf("%d", ptr);
            num2=foo(ptr);
            printf("Number returned: %d\n", num2);
        }

        else if(*ptr==2)
```

```

    {
        num2=foo1(words);
        printf("\nNumber returned:%d\n", num2);

    }

    else if(*ptr==3)
    {
        continue1=0;

    }

    else
    {
        printf("Not a valid entry.\n");
    }

}
}

```

1. The function *foo1* will always return the same value.
2. The initial value of *ptr* is 8.
3. The only way to exit this program is to let *continue1* equal 3.
4. *scanf("%d", ptr);* is the same as *scanf("%d", &num2);*
5. The return value of the function *foo* does not rely on user input.
6. We could change *else if(\*ptr==2)* to *else if(num==2)* and have the same program.
7. For this line: *printf("\n\nEnter a number: 1, 2 or 3 to quit: \n");* the while loop would still run if the user entered 9 since any other value than 0 would be considered as “true”.
8. The functions *foo* and *foo1* have the same number of parameters.
9. For this line: *num2=foo(ptr);* we could pass in *&num* and still have the same result.
10. *&* is the only unary operator used in this code.

## Problem 2 (16 points)-Tiny Code Include with the document Answers.doc

The code you give should be a working code fragment. This means if your code was inserted into a C program, you should not get any errors. If you are expected to give a function declaration or anything else specific, it will be indicated.

**NOTE: YOU CAN SEE SOLVED VERSIONS OF THIS IN THE QUIZ/EXAM INFO I POSTED IN BLACKBOARD UNDER COURSE MATERIALS.**

1. Depending on the temperature and the number of birds, the total number of flowers in Jon’s garden goes up or down. (function declaration)

2. I lost half of my money in the stock market last year.
3. The hotel got a shipment of 50 new beach umbrellas.
4. The number of customers always varies when it rains or the minimum wage falls below \$9.50. (function declaration)

### **Problem 3 (64 points)-Write a program Submit a program called business.c**

---

In a certain country, it is required by law that all businesses follow a certain company structure and pay scale. No fewer than 6 workers are allowed and for every 2 workers there must be a manager. If there is an odd number of workers, then one manager will have 3 workers (but still the same pay as others). For every three managers, there must be a director-the maximum number of managers that can be handled by a director is 5. If there are less than 3 managers (1 or 2, for example), then no director is present.

The ruler of this particular country has asked you to create a program that would allow a business owner to enter a monthly budget, the number of workers and a worker's monthly salary. A director makes five times the amount a worker makes and a manager makes half of what a director makes.

The program should let the business owner know the total that would be spent and if the total spent goes under or over budget. The user should then have the option to rebudget. The problem should only stop when the user does not want to rebudget-it should keep going otherwise. **(see sample run)**

#### **You will do the following:**

1. Given the following main function (see below), define the two functions that are used in the main:
  - a. `float print_out(float* salary, int* num_of_emps);`
  - b. `int rebudget(float* d, float *budget);`
2. Combine the functions you defined in part one with the following main function to create a working program (sample run follows). **NO CREDIT IF YOU MODIFY THE MAIN FUNCTION OR THE FUNCTION DECLARATIONS.**

#### **Main function:**

```
int main()
{

    int i=1;
    float budget, total, salary;
    float *money_ptr=&salary;
    float *total_ptr=&total;
    int employees;

    printf("Enter monthly budget: $");
    scanf("%f", &budget);

    while(i)
    {
```

```

        printf("Enter monthly worker salary: $");
        scanf("%f", &salary);

        printf("Enter total workers:");
        scanf("%d", &employees);

        total=print_out(money_ptr, &employees);
        i=rebudget(total_ptr, &budget);
    }

}

```

### **Sample Run:**

```

-----
Enter monthly budget: $500000
Enter monthly worker salary: $2500
Enter total workers: 11

**Employee info:**
Total workers: 11   Monthly salary: $2500.00
Total managers: 5   Monthly salary: $6250.00
Total directors: 1   Monthly salary: $12500.00
Total spent: $71250.00
--This goes UNDER your budget by $428750.00.

```

```

-----
Would you like to rebudget? 1 for yes 2 for no.
1
Enter monthly worker salary: $5600
Enter total workers:14

```

```

**Employee info:**
Total workers: 14   Monthly salary: $5600.00
Total managers: 7   Monthly salary: $14000.00
Total directors: 2   Monthly salary: $28000.00
Total spent: $232400.00
--This goes UNDER your budget by $267600.00.

```

```

-----
Would you like to rebudget? 1 for yes 2 for no.
2
Bye!

```