Undergraduate Lab Report of Jinan University

Course Title Computer Organization Evaluation

Lab Name Lab 3: Number Storage Instructor SUN Heng

Lab Address N116

Student Name TSUI, Hon Fai Student No 2018058445

College International School

Department Major CST

Date 21 / 10 / 2019 ~~Morning~~ / Afternoon

1. Introduction

1) Objective

This lab extends previous labs. It will immerse you in a problem context, which will require you to apply the theoretical concepts taught in the C language course. This will help you to develop basic skills needed to become a device level system developers and digital system designer. While going through this lab concentrate on developing a good understanding of the problem context, as it is a fundamental requirement for becoming a skilled system designer or application developer.

2) Logistics

Code files are written in gedit and then later compiled in GCC and tested in terminal emulator. An output file is also generated by the program.

2. Instructions

To run this program - first compile to create an executable,
then run the executable code. Open Linux console/shell to compile and make an executable using the gcc compiler.

3. Lab Devices

GCC 4.7 on Ubuntu 12.10 x86 in Lab room

4. Results

```
This program was executed at time : 599785472 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long
lenovo@lenovo:~/桌面$ ./lab3
This program was executed at time : -234881024 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long
lenovo@lenovo:~/桌面$ ./lab3
This program was executed at time : -209715200 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long
```

Image 1 Before Editing

This program was executed at time : -1589641216 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long

Snippet 1 Output file

```
lenovo@lenovo:~/桌面$ ./lab3
This program was executed at time : -1451229184 secs
This program was executed at time : 1571761830.000000 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long
lenovo@lenovo:~/桌面$ gcc -o lab3 lab3.c
```

Image 2 Part (c) Now using matching data-type

This program was executed at time : 1572278683.000000 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long

Snippet 2 Output file for part(c)

```
lenovo@lenovo:~/桌面$ ./lab3
This program was executed at time : 1571761944 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long
```

Image 3 Part(d) Now using *long int*

```
This program was executed at time : 1572278457 secs
The sizes of different data type for this machine and compiler are -
int data type is 4 bytes or 32 bits long
double data type is 8 bytes or 64 bits long
```

Snippet 3 Output file for part(d)

```
lenovo@lenovo:~/OneDrivePersonal/JNU/y2s1/計算機組成/lab/lab3$ ./lab3_2
 unsigned int data type is 4 bytes or 32 bits long
 double data type is 8 bytes or 64 bits long
long data type is 4 bytes or 32 bits long
long long data type is 8 bytes or 64 bits long
 char data type is 1 bytes or 8 bits long
float data type is 4 bytes or 32 bits long
 struct timeval data type is 8 bytes or 64 bits long
```

Image 4 Part(e)

5. Questions

C. When you ran your code, did you get the time executed to be negative? If yes, why did that happen? (Since time cannot be negative). How could you fix this? Figure out how to fix it and do the necessary modifications.

Yes, it is caused by the statement in line 46(also in line 39) where the printf is instructed to take the double floating number time_stamp as input but formatted to output it as %d(integer) instead of %f(floating point). Additionally, on 32bit system, the memory space of double only has 10 digits for significant number. When a large value like the time counting function used here fit into a double variable, there will be a truncation in number without using explicit type conversion, as there are not enough space for saving significant bits, printf read non-numerical bit instead. Hence the error occurs.

To fix it, I changed the printf format back to %lf.

D. Change the data type of the variable time_stamp from double to long int. Is there a change in the values reported? If so, which of the values is the correct value? Why is there a difference?

Yes, the vaule is no longer negative, and being correct.
Instead of using double which uses 8bytes but storing only 10 significant bits, long int has 31significant bits, is sufficient to store such a large number in the situation.

The value reported (using the pair long int, %d) is now consistent with the pair (double, %lf) . As the output no longer have negative value, it is assumed to be correct. The difference comes from the fact that the

values are no longer truncated.

E. Find out the structure of type "timeval". Is it a standard C data type or is it platform specific?

The structure is platform specific. The data type Double take 16bytes on a x86_64 system while 8bytes on x86 system, for instance.
Also, time.h used here is specific to UNIX compiler like GCC, but is non-existent on Windows' MSVC. As UNIX and Windows NT has different mechanism to determine time, the former platform rely on accumulated counting since 1970/01/01 00:00.

6. Discussion

In the experiment, I faced with the limitation of various C data-types.
Mainly is the difference in significant bit and the maximum range of number. Although a double floating no. Data type has a wide ranged representation of no. set, it is unable to store a precise integer larger than 2^10-1.
I would conclude the experiment as an success.

7. Appendix (Program Code)

```
#include <stdio.h> //For input/output
#include <sys/time.h> //For gettimeofday() function

int main()
{
    int int_var; //Tag 1

    struct timeval this_instant;
    double time_stamp;
    //long int time_stamp;


    FILE *my_file_pointer;
    if ( (my_file_pointer = fopen("lab3_out.txt", "w")) == NULL)
    {
        printf("Error opening the file, so exiting\n");
        exit(1);
    }

    gettimeofday(&this_instant,0);
    time_stamp = this_instant.tv_sec;

    //Code segment for file I/O
    fprintf(my_file_pointer, "This program was executed at time : %d secs\n", time_stamp);
    //fprintf(my_file_pointer, "This program was executed at time : %f secs\n", time_stamp);
//Use in part c only
```

```
    fprintf(my_file_pointer, "The sizes of different data type for this machine and compiler are -\n");
    fprintf(my_file_pointer, "int data type is %d bytes or %d bits long\n",sizeof(int_var), sizeof(int_var)*8 );
        fprintf(my_file_pointer, "double data type is %d bytes or %d bits long\n",sizeof(double), sizeof(double)*8 );

    //Code segment for console I/O, this can be used instead of the file I/O
    printf("This program was executed at time : %d secs\n", time_stamp);
//printf("This program was executed at time : %f secs\n", time_stamp);
//Use in part c only

    printf("The sizes of different data type for this machine and compiler are -\n");
    printf("int data type is %d bytes or %d bits long\n",sizeof(int_var), sizeof(int_var)*8 );
//Tag 2
    printf("double data type is %d bytes or %d bits long\n",sizeof(double), sizeof(double)*8 );

printf("The sizes of different data type for this machine and compiler are -\n");
printf("int data type is %d bytes or %d bits long\n",sizeof(int_var), sizeof(int_var)*8 );
//Tag 2

//for lab3_2
printf(" unsigned int data type is %d bytes or %d bits long\n",sizeof(unsigned int), sizeof(unsigned int)*8 );
printf(" double data type is %d bytes or %d bits long\n",sizeof(double), sizeof(double)*8 );
printf("long data type is %d bytes or %d bits long\n",sizeof(long), sizeof(long)*8 );
printf("long long data type is %d bytes or %d bits long\n",sizeof(long long), sizeof(long long)*8 );
printf(" char data type is %d bytes or %d bits long\n",sizeof(char), sizeof(char)*8 );
printf("float data type is %d bytes or %d bits long\n",sizeof(float), sizeof(float)*8 );
printf(" struct timeval data type is %d bytes or %d bits long\n",sizeof(struct timeval), sizeof(struct timeval)*8 );


    fclose(my_file_pointer); //To close the output file, mandatory to actually get an output !

    return 0;
}
```