B E 6th SEMESTER (CST) Final EXAMINATION 2007
Systems Programming (CST 605)
Answer any 5 questions

FM 100 Time: 3 hrs

1. a) DOS interrupt INT 21h function no. 2 implements character output in the default output device where function value is placed in AH register and the code of the character to be printed in DL register. Write a procedure in 80x86 assembly language which prints the content of AX register in decimal. [20]

2. a) What is Macro-processor? Describe its use and advantages.

   b) Assume that you are in a programming environment where it is desirable to save all the registers upon entry to a subroutine and restore them upon exit. Write ENTRY and EXIT macros using 8085A assembly language.

   c) What could be the possible input and output files to and from a linker respectively? What is a compiler driver? What are the phases that a C-program file (say, src.c) goes through when you call *gcc* to create an executable. [6 + 8 + 6]

3. a) Contents of the object file can usually be divided into 5 distinct types. Name them and elaborate the contents in each class.

   b) Draw the diagram showing the different fields of a typical relocatable ELF object file. Describe the fields in short.

   c) Intel 8 bit hex file format is used to produce the object file by many assemblers. Discuss the role played by each field in each record of this file. [6 + 10 + 4]

4. a) Write a program in X86 assembly language to compute the size of the file in bytes; you may keep the byte count in a 16 bit register assuming that the file size is less than 64k. Use the following DOS system calls.

Function. No. 0Ah Buffered Keyboard Input

On entry	Register Content
AH	0AH
DS:DX	Pointer to an input buffer
On Return	Register Content
	None

The first byte of the input buffer specifies the number of characters the buffer can hold. The second byte is set to the number of characters actually read excluding the carriage return which is always the last character.

Function. No. 3DH Open a file

On entry	Register Content
AH	3DH
DS:DX	Pointer to an ASCIIZ path name
AL	Access Code (e.g.; 0 = read)
On Return	**Register Content**
AX	16 bit file handle (CF=0) error code if CF = 1

Function. No. 3FH Read a file

On entry	Register Content
AH	3FH
BX	File handle
DS:DX	Buffer address
CX	Number of bytes to be read
On Return	**Register Content**
AX	No. of bytes read (CF=0) error code if CF = 1

b) Using appropriate DOS interrupt calls write a tool for DOS environment to display the contents of any binary file. Note that for simplicity you may read the file to be displayed using I/O redirection technique. You are allowed to use DOS interrupt INT 21h function no. 7 and function no. 2 only (which are logical equivalent to getchar() and putchar(c)).

Note that function 02H requires that ASCII code of the character to be printed (in default output device) in DL register and function 7 returns the byte read from the default input device in AL register.

Print the bytes in Hex with 16 bytes per line. You need not print the address offset in each line. [13 + 7]

5. a) Consider a pair of C language source files m.c and a.c.

```
/*  m.c  */
extern void ax(char *); int main(int ac, char **av)
{
static char string[] = "Hello, world!\n";
prn(string);
}
```

```
/*  a.c  */
#include <unistd.h>
#include <stdio.h>
void prn(char *s)
{
write(1, s, strlen(s));
}
```

Find the relocation entries and the symbols to be exported and imported.

b) The following table shows different sections and their sizes of an executable object file:

Name	Text	Data	bss
main	1502	302	65
ProcA	498	317	101
ProcB	1300	165	712
ProcC	256	1805	220

(All the data are in HEX)

Considering 8 byte word alignment and a page size of 0x1000 bytes show the run time layout with the load point as 0x2000. [8 + 12]

6. Create a static library named libcmplx.a which contains two member functions, namely, addcmplx.o and mulcmplx.o. As the names suggest the member functions compute elementary operations; addition and multiplication of two complex numbers and return the result to a third variable of same type. Write the functions. Also, you are to write a main program (main3.c) to test the static library functions for the complex numbers $10 + i15$ and $7 + i9$.

   Show the command lines for creating the .o files, .a files and linking of main.o with the static library. [20]

7. a) With a neat diagram show how the ELF format exhibits linkable sections and loadable segments simultaneously.

   b) Describe in detail the structure of UNIX a.out object file.

   c) Explain what extra information the assembler must provide so as to enable separate assembly modules to be linked by the linker. Use suitable example in your answer.

   d) Consider the following program, which consists of two modules

```
/* x.c  */
void p2(void)

int main()
{
p2();
return(0);
}
```

```
/* b.c */

#include <stdio.h>
char main;
void p2()
{
printf("0x%x\n", main);
}
```

   This program, even though p2 never initializes variable main, prints the string

```
"0x55\n"
```

   and terminates normally, – explain? [4 + 5 + 4 + 7])

8. Write short notes on (Any two) [2 × 10]
   (i) Expression evaluation in assembler; problem and solution.
   (ii) PIC in ELF.
   (iii) Functions for dynamic loading and linking from applications.