Roll No: 1803070

Lab Performance Test [1]

Lab Task Q[No 1]

Q1 **Question:**

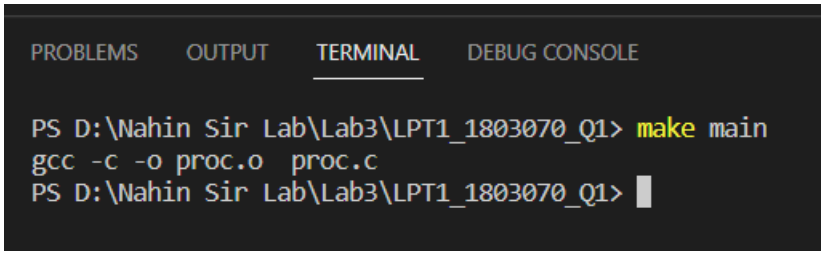
Consider given code, #include<math.h> int main(){ float a=1; float b=2; float c=a+b; return 0; } . Show output file with extension ".o" generated by C compiler along with Makefile (point penalty for adding extra commands other than necessary).

Solution (Bold your own written code):

Makefile:

```
main:
    gcc -c -o proc.o  proc.c
```

Output (Screen/SnapShot):



Lab Task Q[No 2a]

Q2a **Question:**

Consider given statements, 1. We Like to Buy 10 Apples 2. I Like to Buy 20 Oranges 3. We Want to Buy 30 Mangoes . a) Show a flex file which can tokenize given statements. b) Show a bison file which can parse given statements.

Solution (Bold your own written code):

.l file:

```
%option noyywrap

%{
%}

%%
"We"|"I" {printf("%s -> Person\n", yytext);}
"Like"|"Want" {printf("%s -> Verb\n",yytext);}
```

```
"to" {printf("%s -> To\n",yytext);}
"Buy" {printf("%s -> Buy\n",yytext);}
"10"|"20"|"30" {printf("%s -> Amount\n",yytext);}
"Apples"|"Mangoes" {printf("%s -> Fruits\n",yytext);}
. {}
%%

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

Makefile:

```
flex2:
    flex 2.1
    gcc lex.yy.c
    ./a.exe < input2.txt
```

Output (Screen/SnapShot):

```
PS D:\Nahin Sir Lab\Lab3\LPT1_1803070_Q2a> make flex2
flex 2.1
gcc lex.yy.c
./a.exe < input2.txt
We -> Person
Like -> Verb
to -> To
Buy -> Buy
10 -> Amount
Apples -> Fruits

I -> Person
Like -> Verb
to -> To
Buy -> Buy
20 -> Amount
Apples -> Fruits

We -> Person
Want -> Verb
to -> To
Buy -> Buy
30 -> Amount
Mangoes -> Fruits
PS D:\Nahin Sir Lab\Lab3\LPT1_1803070_Q2a>
```

Lab Task Q[No 2b]

Q2a **Question:**

Consider given statements, 1. We Like to Buy 10 Apples 2. I Like to Buy 20 Oranges 3. We Want to Buy 30 Mangoes . b) Show a bison file which can parse given statements.

Solution (Bold your own written code):

.l file:

```
%option noyywrap

%{
    //1803070
#include "prog1.tab.h"
%}

%%
"We"|"I" {return(N);}
"Like"|"Want" {return(V);}
"to" {return(P);}
"Buy" {return(VV);}
"10"|"20"|"30" {return(A);}
"Apples"|"Mangoes" {return(O);}
%%
```

.y file:

```
%{
    //1803070
#include<stdio.h>
void yyerror(char *s);
int yylex();
%}

%token N V P VV A O
%start S

%%
S: N V P VV A O;
%%

int main()
{
    if(!yyparse())
        printf("ACCEPTED\n");
    return 0;
```

```
}

void yyerror(char *s)
{
    fprintf(stderr, "error: %s\n", s);
}
```

Makefile:

```
prog1:
    bison -d prog1.y
    flex prog1.l
    gcc prog1.tab.c lex.yy.c
    a.exe <input.txt
```

Output (Screen/SnapShot):

```
PS D:\Nahin Sir Lab\Lab3\LPT1_1803070_Q2b> make prog1
bison -d prog1.y
flex prog1.l
gcc prog1.tab.c lex.yy.c
a.exe <input.txt
     ACCEPTED
PS D:\Nahin Sir Lab\Lab3\LPT1_1803070_Q2b>
```