Name: Mohammad Awais
Class: BSCS-8-A
CMS: 242554

Compiler Construction Lab 9

Task 1

< Code >

```
#include <iostream>
using namespace std;


#include <vector>
#include<string>
```

```
#include <typeinfo>
```

```
/******************************** DECLARATIONS *****************************/
```

```
string input;
```

```
vector<char> num_list = {’0’,’1’,’2’,’3’,’4’,’5’,’6’,’7’,’8’,’9’,’.’};
```

```
int main(){
cout<<"\n\t ********** TASK 1 ************* \n\n";
```

```
cout<<"[?] Enter any value : ";
getline(cin, input);
int mode = -1; // 0 : STRING, 1: INT, 2: FLOAT, 3: DOUBLE
```

```
for(char c : input){
int num_check = 0;
for(char n : num_list){
if (c == n){
num_check = 1;
if(n == ’.’ && mode !=2){
mode =2;
}
else if(n == ’.’ && mode == 2){
num_check = 0 ;
}
break;
}
}
```

```
if(num_check == 0){
mode=0;
break;
}
}
```

```
if(mode == 0){
string value = input;
```

```
cout<<"\n\t[+] Value Entered : "<<value<<endl;
cout<<"\n\t[+] Value Type : String ("<<typeid(value).name()<<")"<<endl;
}
else if (mode != 2){
int value = stoi(input);
```

```
cout<<"\n\t[+] Value Entered : "<<value<<endl;
cout<<"\n\t[+] Value Type : Int ("<<typeid(value).name()<<")"<<endl;
}
else{
string partA = "";
```

```
string partB = "";
int bCounter = 0;
```

```
int gear = 0;
for(char c : input){
if(gear == 0){
if(c == ’.’){
gear =1;
continue;
}
partA = partA + c;
}
else if ( gear == 1){
partB = partB + c;
bCounter++;
}
}
```

```
if (partA.length()==0){
partA+=’0’;
}
```

```
if (partB.length()==0){
```

```
partB+=’0’;
}
```

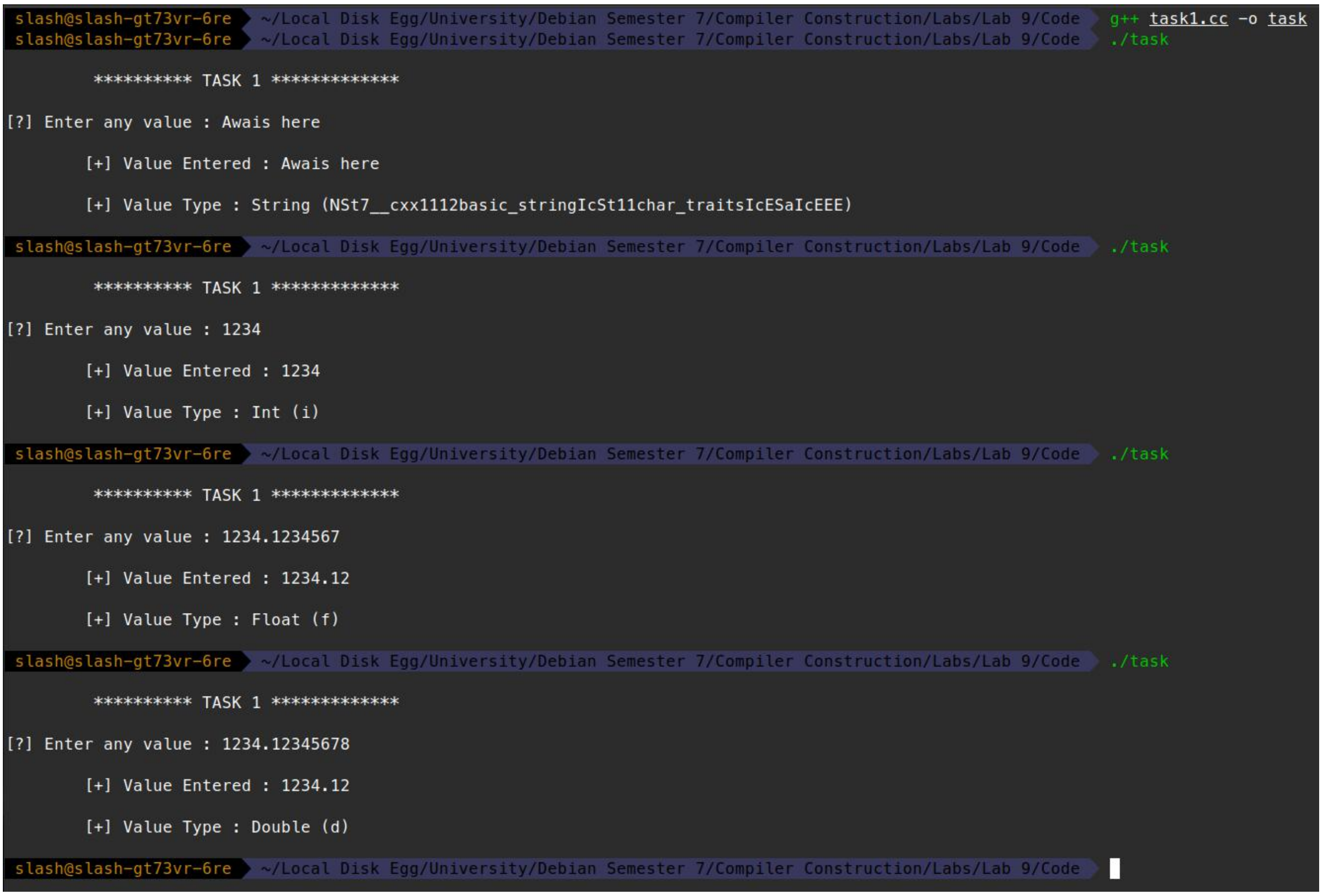
```
if(bCounter > 7){
double value = stod(partA+"."+partB);
```

```
cout<<"\n\t[+] Value Entered : "<<value<<endl;
cout<<"\n\t[+] Value Type : Double ("<<typeid(value).name()<<")"<<endl;
}
else{
float value = stof(partA+"."+partB);
```

```
cout<<"\n\t[+] Value Entered : "<<value<<endl;
cout<<"\n\t[+] Value Type : Float ("<<typeid(value).name()<<")"<<endl;
}
}
```

```
cout<<endl;
}
```

< Output >



```
slash@slash-gt73vr-6re  ~/Local Disk Egg/University/Debian Semester 7/Compiler Construction/Labs/Lab 9/Code  g++ task1.cc -o task
slash@slash-gt73vr-6re  ~/Local Disk Egg/University/Debian Semester 7/Compiler Construction/Labs/Lab 9/Code  ./task

         ********** TASK 1 *************

[?] Enter any value : Awais here

        [+] Value Entered : Awais here

        [+] Value Type : String (NSt7__cxx1112basic_stringIcSt11char_traitsIcESaIcEEE)
slash@slash-gt73vr-6re  ~/Local Disk Egg/University/Debian Semester 7/Compiler Construction/Labs/Lab 9/Code  ./task

         ********** TASK 1 *************

[?] Enter any value : 1234

        [+] Value Entered : 1234

        [+] Value Type : Int (i)
slash@slash-gt73vr-6re  ~/Local Disk Egg/University/Debian Semester 7/Compiler Construction/Labs/Lab 9/Code  ./task

         ********** TASK 1 *************

[?] Enter any value : 1234.1234567

        [+] Value Entered : 1234.12

        [+] Value Type : Float (f)
slash@slash-gt73vr-6re  ~/Local Disk Egg/University/Debian Semester 7/Compiler Construction/Labs/Lab 9/Code  ./task

         ********** TASK 1 *************

[?] Enter any value : 1234.12345678

        [+] Value Entered : 1234.12

        [+] Value Type : Double (d)
slash@slash-gt73vr-6re  ~/Local Disk Egg/University/Debian Semester 7/Compiler Construction/Labs/Lab 9/Code
```

Task 2

< Code >

Scope.y

```
%{
#include <string>

#include <iostream>
```

```
#include <vector>
#include <array>
```

```
#include <math.h>
```

```
#define YYDEBUG 1
```

```
void Push();
std::vector<std::vector<int>> Pop();
std::vector<std::vector<int>> Top();
```

```
std::vector<std::string> id_list = {};
```

```
void newIdentifier(std::string id_name, int id_value);
void Identifier(std::string id_name, int id_value);
int returnIdentifier(std::string id_name);
```

```
int yylex();
void yyerror();
extern FILE * yyin;
```

```
%}
```

```
%union {
int val;
char *name;
};
```

```
%token keyword_int
%token T_int
%token identifier
%token cout
```

```
%%
S : E { };
```

```
E : keyword_int identifier '=' T_int ';' {newIdentifier(yylval.name, yylval.val);}
| keyword_int identifier';' {newIdentifier(yylval.name, NULL);}
| identifier '=' T_int ';' {Identifier(yylval.name, yylval.val);}
| cout '<''<' identifier ';'{std::cout<<">> "<<yylval.name<<" = "<<returnIdentifier(yylval.name)<<std::endl;}
| '{' E '}' { Push();}
| '}' E {Pop();}
```

```
%%
```

```
const int MAXSIZE = 40;
static std::array<std::vector<std::vector<int>>,MAXSIZE> stack;
```

```
void Push(){
```

```
std::vector<std::vector<int>> data = {};
```

```
for (int i = MAXSIZE-1; i>0;i--){
stack[i]=stack[i-1];
}
stack[0] = data;
}
```

```
std::vector<std::vector<int>> Pop(){
std::vector<std::vector<int>> data = stack[0];
for (int i = 0; i<MAXSIZE;i++){
stack[i]=stack[i+1];
}
return data;
}
```

```
std::vector<std::vector<int>> Top(){
return stack[0];
}
```

```
void newIdentifier(std::string id_name, int id_value){
int detect = 0;
int target_id = -1;
for( int i = 0; i<id_list.size() ; i++){
if(id_name==id_list[i]){
detect = 1;
target_id = i;
break;
}
}
```

```
if(detect == 0){
id_list.push_back(id_name);
target_id = id_list.size();
}
```

```
detect = 0;
for ( int i = 0; i<stack[0].size(); i++){
if(stack[0][i][0]==target_id){
detect = 1;
break;
}
}
```

```
if(detect==1){
// ERROR BECAUSE TWO DECLARATIONS IN SAME SCOPE
std::cout<< " GIVE ERROR YOU FOOL [NEW]";
}
else{
```

```
stack[0].push_back({target_id,id_value});
}
}
```

```
void Identifier(std::string id_name, int id_value){
int detect = 0;
int target_id = -1;
for( int i = 0; i<id_list.size() ; i++){
if(id_name==id_list[i]){
detect = 1;
target_id = i;
break;
}
}
```

```
if(detect == 0){
// ERROR BECAUSE TWO DECLARATIONS IN SAME SCOPE
std::cout<< " GIVE ERROR YOU FOOL [ SIMPLE ]";
}
```

```
detect = 0;
int detect_j = -1;
int detect_i = -1;
for(int j = 0; j<stack.size();j++){
for ( int i = 0; i<stack[j].size(); i++){
if(stack[j][i][0]==target_id){
detect = 1;
detect_j = j;
detect_i = i;
break;
}
}
if(detect == 1){
break;
}
}
```

```
if(detect==1){
stack[detect_j][detect_i][1]=id_value;
}
else{
// ERROR BECAUSE TWO DECLARATIONS IN SAME SCOPE
std::cout<< " GIVE ERROR YOU FOOL [SIMPLE 2]";
}
}
```

```
int returnIdentifier(std::string id_name){
int detect = 0;
int target_id = -1;
```

```
for( int i = 0; i<id_list.size() ; i++){
if(id_name==id_list[i]){
detect = 1;
target_id = i;
break;
}
}
```

```
if(detect == 0){
// ERROR BECAUSE TWO DECLARATIONS IN SAME SCOPE
std::cout<< " GIVE ERROR YOU FOOL [ RETURN SIMPLE ]";
}
```

```
detect = 0;
int detect_j = -1;
int detect_i = -1;
for(int j = 0; j<stack.size();j++){
for ( int i = 0; i<stack[j].size(); i++){
if(stack[j][i][0]==target_id){
detect = 1;
detect_j = j;
detect_i = i;
break;
}
}
if(detect == 1){
break;
}
}
```

```
if(detect!=1){
// ERROR BECAUSE TWO DECLARATIONS IN SAME SCOPE
std::cout<< " GIVE ERROR YOU FOOL [RETRUN SIMPLE 2]";
}
```

```
return stack[detect_j][detect_i][1];
}
```

```
int main() {
//return yyparse();
```

```
FILE *file = fopen("testcode","r");
if(!file){
printf("[-] Can't open the file!\n");
return -1;
}
yyin=file;
//while(yyparse());
do
{
```

```
yyparse();
} while (!feof(yyin));
fclose(file);
printf("\n");
```

```
return 0;
}
```

Scope.l

```
%{
#include "scope.tab.h"
%}



%%
```

```
[0-9]+ { yylval.val = atoi(yytext); return T_int;}
[=;{}<] { return yytext[0];}
[i][n][t] { return keyword_int;}
[c][o][u][t] {return cout;}
[a-zA-Z_][a-zA-Z_0-9]* { yylval.name = yytext; return identifier;}
. { /* ignore everything else */ }
```

```
%%
```

```
/*
int main(int argc, char **argv)
{
FILE *file = fopen("testcode","r");
if(!file){
printf("[-] Can't open the file!\n");
return -1;
}
yyin=file;
while(yylex());
fclose(file);
printf("\n");
}*/
```

< Output >