

BHASH language Sample Programs

1. Program to find the factorial of the given number

```
func main():
```

```
int a = input("Enter a number");
```

```
int b = factorial(a);
```

```
print(b);
```

```
end
```

```
func factorial(int x):
```

```
int temp=1;
```

```
for(int i = 1 : x : -1) :
```

```
temp = temp*i ;
```

```
end
```

```
return temp;
```

```
end
```

```
manikanta@manikanta: ~/Desktop
manikanta@manikanta:~/Desktop$ yacc -d bhash latest.y
bhash_latest.y: warning: 17 shift/reduce conflicts [-Wconflicts-sr]
bhash_latest.y: warning: 9 reduce/reduce conflicts [-Wconflicts-rr]
manikanta@manikanta:~/Desktop$ lex bhash.l
manikanta@manikanta:~/Desktop$ cc lex.yy.c y.tab.c -obas.exe -lm
y.tab.c: In function 'yyparse':
y.tab.c:1466:16: warning: implicit declaration of function 'yylex' [-Wimplicit-function-declaration]
    yychar = yylex ();
                   ^
manikanta@manikanta:~/Desktop$ ./bas.exe
parsed successfully
manikanta@manikanta:~/Desktop$ {"parsed successfully\n");}
;
declarationlist: declaration declarationlist
                | declaration
                ;
declaration : function_definition
            | variable_declaration
            ;
function_definition : FUNC ID '(' params ')' statement;
variable_declaration : variable_declaration datatype init_dec_list
                    | datatype init_dec_list
                    ;
init_dec_list : init_dec ';' ;
```

2. Finding the maximum and minimum and returning multiple values from function

func main():

int a,b,c;

a = input("Enter first number");

b = input("Enter second number");

c = input("Enter third number");

int max,min = maxMin(a,b,c);

print(max,min);

end

func maxMin(int x , int y, int z):

int max = x,min = x;

if(y>max):

max = y;

end

if(z>max):

max = z;

end

if(y<min):

min = y;

end

if(z<min):

min = z;

end

return max,min;

end

```
manikanta@manikanta: ~/Desktop
manikanta@manikanta:~/Desktop$ yacc -d bhash_latest.y
bhash_latest.y: warning: 17 shift/reduce conflicts [-Wconflicts-sr]
bhash_latest.y: warning: 9 reduce/reduce conflicts [-Wconflicts-rr]
manikanta@manikanta:~/Desktop$ lex bhash.l
manikanta@manikanta:~/Desktop$ cc lex.yy.c y.tab.c -obas.exe -lm
y.tab.c: In function 'yyparse':
y.tab.c:1466:16: warning: implicit declaration of function 'yylex' [-Wimplicit-function-declaration]
    yychar = yylex ();
                  ^
manikanta@manikanta:~/Desktop$ ./bas.exe
parsed successfully
manikanta@manikanta:~/Desktop$ ("parsed successfully\n");
;
declarationlist: declaration declarationlist
                | declaration
                ;
declaration : function_definition
            | variable_declaration
            ;
function_definition : FUNC ID '(' params ')' statement;
variable_declaration : variable_declaration datatype init_dec_list
                    | datatype init_dec_list
                    ;
init_dec_list : init_dec ';' ;
```

3. Labeled Blocks

```
func main():
    block outerLoop;
    int n,length;
    n = input("Enter number of test cases");
    length = input("Enter the number of inputs
per testcase");
```

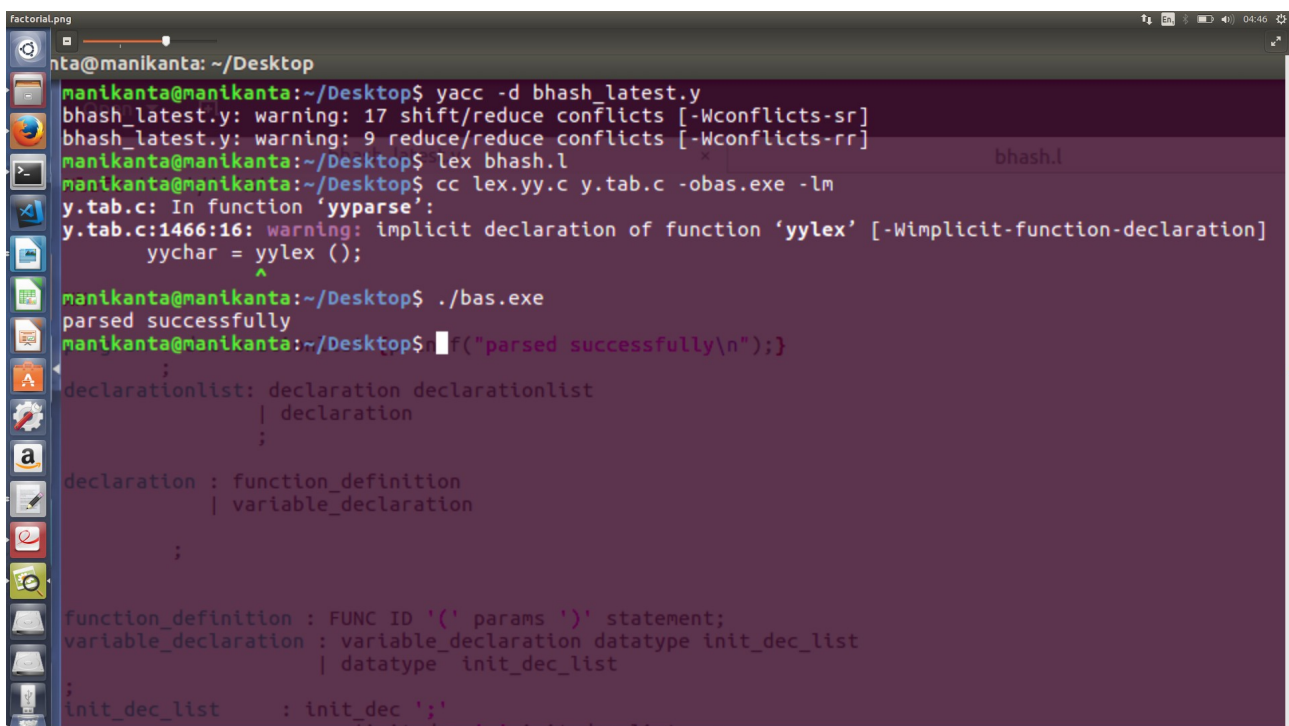
```
outerLoop :
    for(int i=0 : n : 1):
        for(int j=0 : length : 1):

            defvar a;
```

```

a = input();
string b = typeDef(a);
if(b=="int"):
    println("You have entered wrong
output");
    break outerLoop;
end
end
end
end
end
end

```



```

manikanta@manikanta: ~/Desktop
manikanta@manikanta:~/Desktop$ yacc -d bhash_latest.y
bhash_latest.y: warning: 17 shift/reduce conflicts [-Wconflicts-sr]
bhash_latest.y: warning: 9 reduce/reduce conflicts [-Wconflicts-rr]
manikanta@manikanta:~/Desktop$ lex bhash.l
manikanta@manikanta:~/Desktop$ cc lex.yy.c y.tab.c -obas.exe -lm
y.tab.c: In function 'yyparse':
y.tab.c:1466:16: warning: implicit declaration of function 'yylex' [-Wimplicit-function-declaration]
    yychar = yylex ();
                   ^
manikanta@manikanta:~/Desktop$ ./bas.exe
parsed successfully
manikanta@manikanta:~/Desktop$ printf("parsed successfully\n");
;
declarationlist: declaration declarationlist
                | declaration
                ;
declaration : function_definition
            | variable_declaration
            ;
function_definition : FUNC ID '(' params ')' statement;
variable_declaration : variable_declaration datatype init_dec_list
                    | datatype init_dec_list
;
init_dec_list : init_dec ':'

```

