Unit 8 Assignment 1

Name: Spuritha Mudireddy

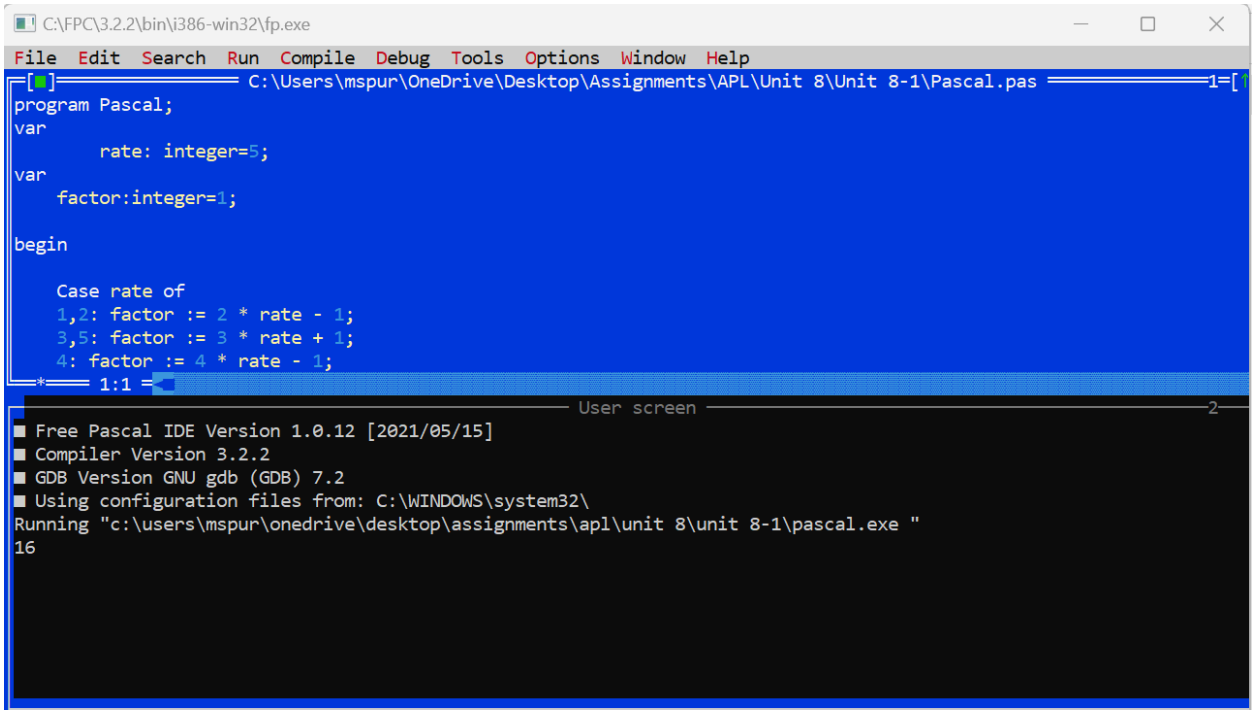
CSULB ID: 030743269

Rewrite the following code segment using a multiple-selection statement in the following languages:

if `((rate == 1) || (rate == 2)) factor = 2 * rate - 1`
if `((rate == 3) || (rate == 5)) factor = 3 * rate + 1`
if `(rate == 4) factor = 4 * rate - 1`
if `((rate == 6) || (rate == 7) || (rate == 8)) factor = rate – 2`
else factor = rate

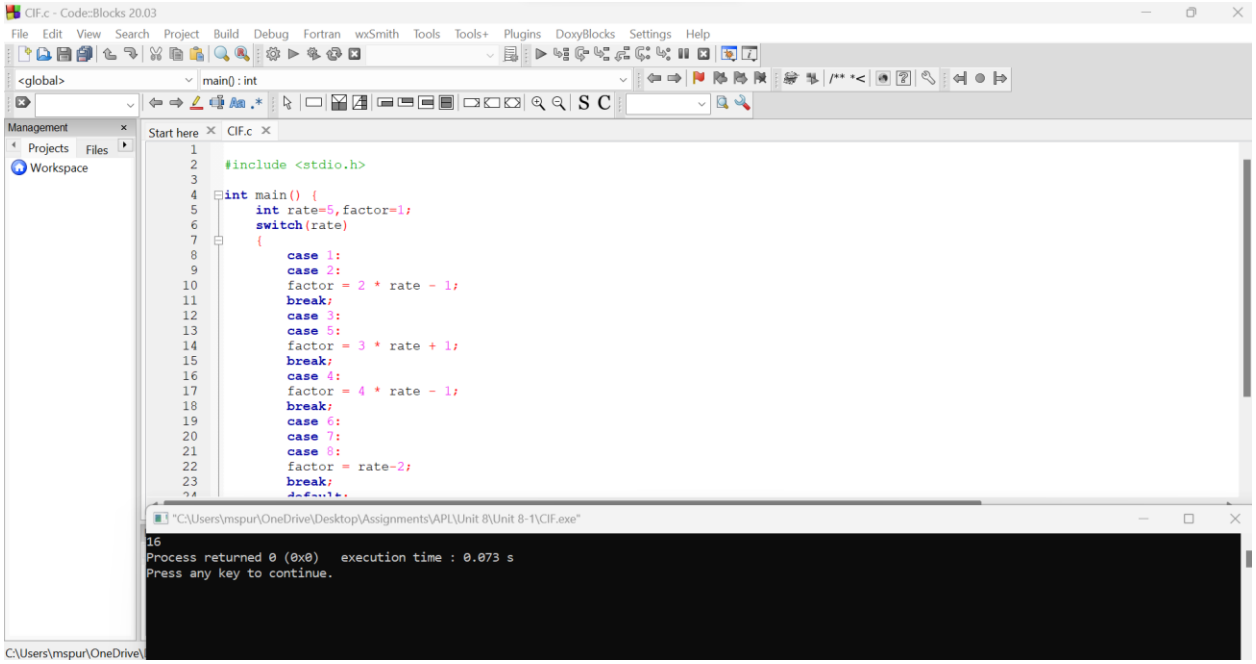
Pascal

```
program Pascal;
var
      rate: integer=5;
var
    factor:integer=1;

begin

    Case rate of
    1,2: factor := 2 * rate - 1;
    3,5: factor := 3 * rate + 1;
    4: factor := 4 * rate - 1;
    6,7,8: factor := rate-2;
      else
            factor:=rate;
    end;
    writeln(factor)

end.
```


C:\FPC\3.2.2\bin\i386-win32\fp.exe
File Edit Search Run Compile Debug Tools Options Window Help
C:\Users\mspur\OneDrive\Desktop\Assignments\APL\Unit 8\Unit 8-1\Pascal.pas
program Pascal;
var
rate: integer=5;
var
factor:integer=1;
begin
Case rate of
1,2: factor := 2 * rate - 1;
3,5: factor := 3 * rate + 1;
4: factor := 4 * rate - 1;
1:1
User screen
Free Pascal IDE Version 1.0.12 [2021/05/15]
Compiler Version 3.2.2
GDB Version GNU gdb (GDB) 7.2
Using configuration files from: C:\WINDOWS\system32\
Running "c:\users\mspur\onedrive\desktop\assignments\apl\unit 8\unit 8-1\pascal.exe "
16

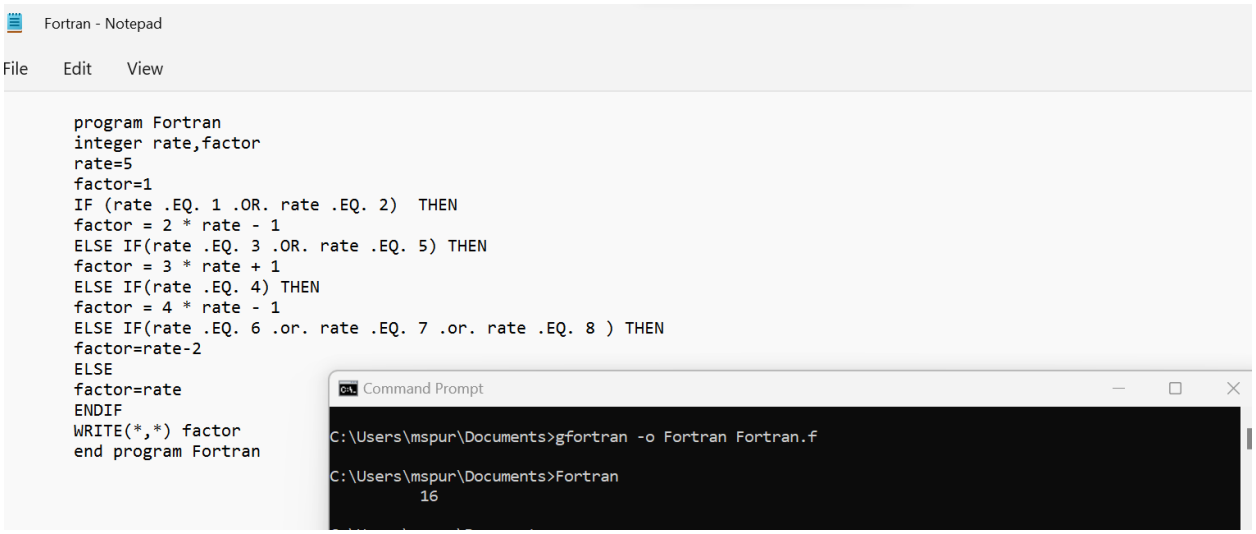
b. C

```
#include <stdio.h>
 int main() {
 int rate=5,factor=1;
 switch(rate)
 {
    case 1:
    case 2:
    factor = 2 * rate - 1;
    break;
    case 3:
    case 5:
    factor = 3 * rate + 1;
    break;
    case 4:
    factor = 4 * rate - 1;
    break;
    case 6:
    case 7:
    case 8:
    factor = rate-2;
    break;
    default:
    factor = rate;
    break;
 }
printf("%d",factor);
 return 0;
}
```


CIF.c - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global>
main() : int
Management
Projects
Files
Workspace
Start here
CIF.c
#include <stdio.h>
int main() {
int rate=5,factor=1;
switch(rate)
{
case 1:
case 2:
factor = 2 * rate - 1;
break;
case 3:
case 5:
factor = 3 * rate + 1;
break;
case 4:
factor = 4 * rate - 1;
break;
case 6:
case 7:
case 8:
factor = rate-2;
break;
"C:\Users\mspur\OneDrive\Desktop\Assignments\APL\Unit 8\Unit 8-1\CIF.exe"
16
Process returned 0 (0x0) execution time : 0.073 s
Press any key to continue.
C:\Users\mspur\OneDrive\

FORTRAN 77 (not one minute later)

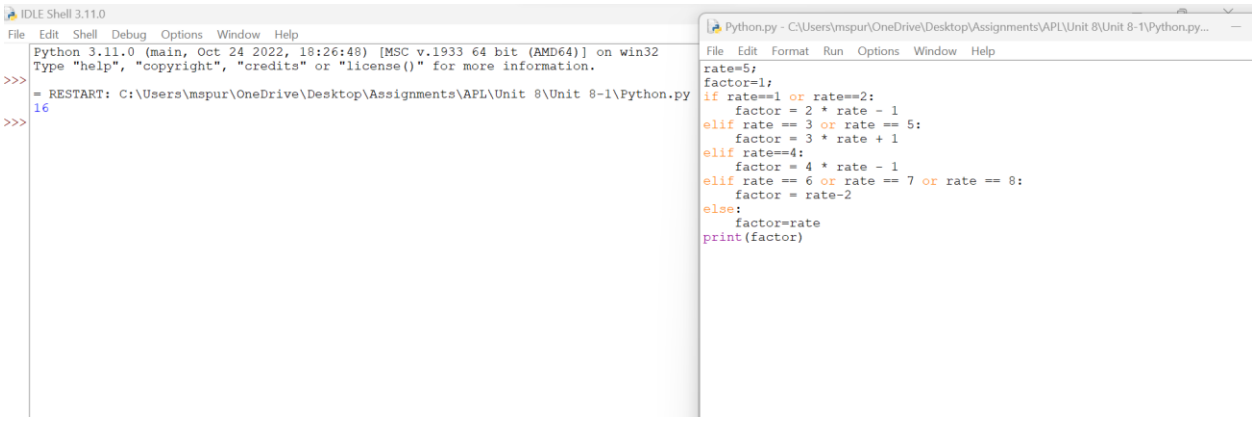
```
      program Fortran
       integer rate,factor
       rate=5
       factor=1
       IF (rate .EQ. 1 .OR. rate .EQ. 2)  THEN
      factor = 2 * rate - 1
       ELSE IF(rate .EQ. 3 .OR. rate .EQ. 5) THEN
      factor = 3 * rate + 1
       ELSE IF(rate .EQ. 4) THEN
      factor = 4 * rate - 1
       ELSE IF(rate .EQ. 6 .or. rate .EQ. 7 .or. rate .EQ. 8 ) THEN
      factor=rate-2
      ELSE
      factor=rate
       ENDIF
      WRITE(*,*) factor
       end program Fortran
```


Fortran - Notepad
File Edit View
program Fortran
integer rate,factor
rate=5
factor=1
IF (rate .EQ. 1 .OR. rate .EQ. 2) THEN
factor = 2 * rate - 1
ELSE IF(rate .EQ. 3 .OR. rate .EQ. 5) THEN
factor = 3 * rate + 1
ELSE IF(rate .EQ. 4) THEN
factor = 4 * rate - 1
ELSE IF(rate .EQ. 6 .or. rate .EQ. 7 .or. rate .EQ. 8) THEN
factor=rate-2
ELSE
factor=rate
ENDIF
WRITE(*,*) factor
end program Fortran
Command Prompt
C:\Users\mspur\Documents>gfortran -o Fortran Fortran.f
C:\Users\mspur\Documents>Fortran
16

d. Python

```
rate=5;
factor=1;
if rate==1 or rate==2:
    factor = 2 * rate - 1
elif rate == 3 or rate == 5:
    factor = 3 * rate + 1
elif rate==4:
    factor = 4 * rate - 1
elif rate == 6 or rate == 7 or rate == 8:
    factor = rate-2
else:
    factor=rate
print(factor)
```

Output:



e. Ada

```
with Ada.Text_IO; use Ada.Text_IO;

procedure Hello is

begin

   declare

     rate : Integer := 5;factor : Integer := 1;

   begin

      case rate is

      when 1|2 =>

         factor := 2 * rate - 1;

      when 3|5 =>

         factor := 3 * rate + 1;

      when 4 =>

         factor := 4 * rate - 1;

      when 6|7|8 =>

         factor := rate - 2;

      when others =>

         factor := rate;


      end case;

   Ada.Text_IO.Put_Line (Integer'Image (factor));

   end;


end Hello;
```

```
gcc -c hello.adb
gnatbind -x hello.ali
gnatlink hello.ali -o hello
16
```