25-10-2021 COMPILER DESIGN SACHIN RAGHUL T

2019103573

OBSERVATION - 07

1. To conver an if statement multiple conditions to an nested-if statement using pattern-action.

Code :-

```
%option noyywrap
%{
   #include <stdio.h>
   int i, n=0, j;
%}
%%
if\(.+\) {
   i=0;
   while (yytext[i] != '&' && i<yyleng) {
     printf("%c", yytext[i]);
     i++;
   }
   printf(")\n");
   n++;
   while (i < yyleng) {
     i+=2;
     if (i > yyleng)
       break;
     for(j=0; j<n; j++)
       printf("   ");
     printf("if(");
     while (yytext[i] != '&' && i<yyleng && yytext[i] != ')') {
       printf("%c", yytext[i]);
       i++;
     }
     n++;
     Printf(")\n");
   }
}
```

```
.*; {
    for(i=1; i<n; i++)
        printf(" ");
    printf("%s", yytext);
}
.|\n
%%
void main()
{
    yyin = fopen("input71.txt", "r");
    yylex();
}
```

input.txt

```
if( x == 5 && y == 10 && z == 15)
    Statement;
```

Output

```
if(x==5)
  if y==10)
     if(z==15)
        Statement;
```

2. convert nested if-else to nested do-while syntax using LEX.

Code :-

```
%option noyywrap
%{
  #include<string.h>
  int n=0, i, j, m;
  char stack[10][1024];
%}
%%
if \(.+\) {
  printf("while(");
  j=0;
  for(i=3; i<yyleng-1; i++){
    printf("%c", yytext[i]);
    if(yytext[i] == '=' && yytext[i+1] == '=')
      stack[n][j++] = '!';
    else
      stack[n][j++] = yytext[i];
  }
  printf(") {");
  n++;
}
.*; {
  for(i=0; i<yyleng && yytext[i] == ' '; i++);
  m = i/4;
  printf("%s\n", yytext);
  for(i=0; i<m; i++)
    printf(" ");
  printf("break.\n");
  for(i=0; i<m-1; i++)
    printf(" ");
```

```
    printf(" }");
    n--;
}
*else {
    for(i=0; yyleng && yytext[i]==' '; i++);
    m = i/4;
    for(i=m; i<n; i++) {
        for(j=0; j<n-i; j++) printf(" ");
        for(j=1; j<n-i; j++) printf(" ");
        printf(" }\n");
    }
    n = m;
    for(i=0; i<m; i++) printf(" ");
    printf("while(%s) {", stackEnt);
    n++;
}
%%
void main()
{
    FILE *fd = fopen("ip72.txt", "r");
    yyin = fd;
    yylex();
    printf("\n");
    for(i=0; i<n; i++) {
        for(j=0; j<n-i; j++) {
            printf(" ");
        }
        printf("break;\n");
        for(j=1; j<n-i; j++) {
            printf(" ");
        }
        printf("}\n");
    }
}
```

input.txt

```
if (x==1)
  if (y==2)
    statement 1;
  else
    statement 2;
else
 if(z==3)
    statement 3;
  else
    statement 4;
```

output:

```
while (x==1){
  while(y==2){
    statement 1;
    break;
  }
  while (y!=2){
    statement 2;
    break;
  }
  break;
}
while (x!=1){
  while(z==3){
    statement 3;
    break;
  }
  while(z!=3){
    statement 4;
    break;
  }
  break;
}
```

3. convert a nested if-else if considering of multiple expressions into a single if-else if.

Code :-

```
%option noyywrap
%{
  #include <stdio.h>
  int i, n=0, j;
%}
%%
if \(.+\) {
  if(n==0) {
    for(i=0; i<yyleng-1; i++)
      printf("%c", yytext[i]);
    n++;
  }
  else {
    printf(" && ");
    for(i=3; i<yyleng-1; i++)
      printf("%c", yytext[i]);
    n++;
  }
}
else if \(.+\) {
  n=1;
  printf("\n");
  for(i=0; i<yyleng-1; i++)
    printf("%c", yytext[i]);
}
```

```
.* : {
    printf(" )\n");
    i=0;
    while(yytext[i]==' ' && i<yyleng)
      i++;
    printf(" ");
    while(i<yyleng){
      printf("%c",yytext[i]);
      i++;
    }
}
.|\n
%%
void main()
{
   yyin=fopen("inp.txt","r");
   yylex();
}
```

input.txt

```
if (a==5 || e==5)
  if(b==2)
     statement 1;
else if (c==3)
  if(d==4)
     statement 2;
```

output:

```
if (a==1 || e==5 && b==2)
  statement 1;
else if (c==3 && d==4)
  statement 2;
```