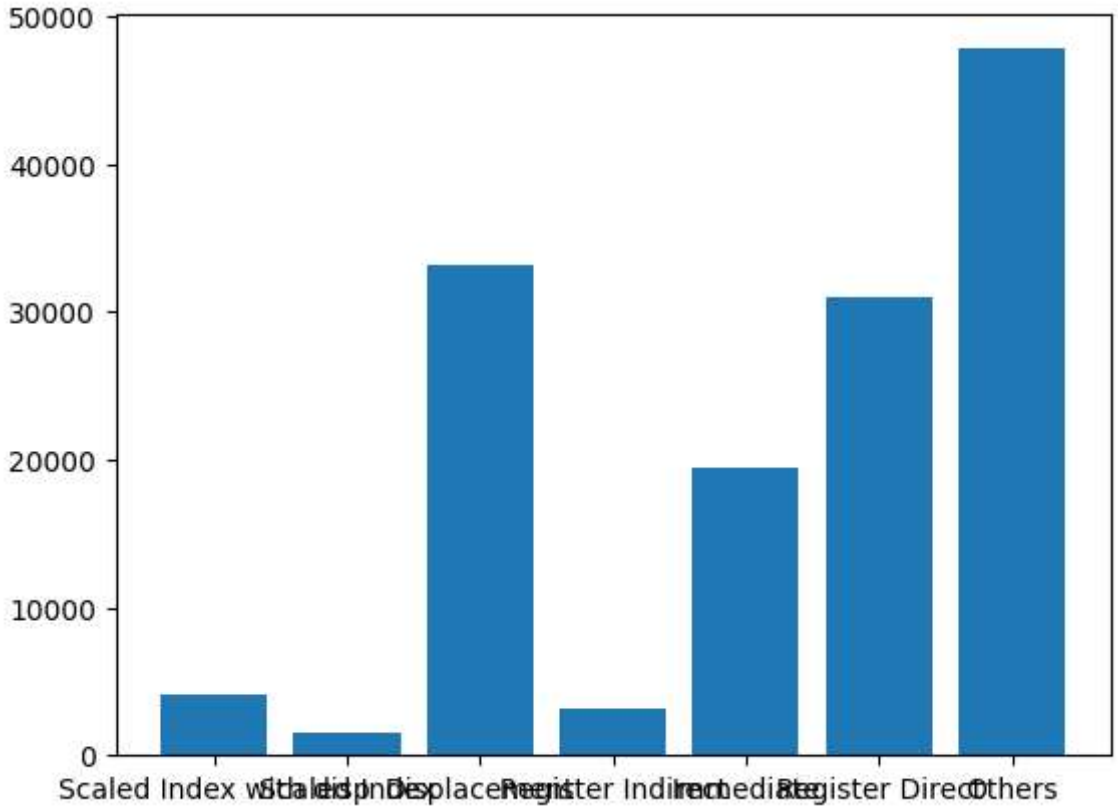
Niraj Nagrale CS22M109

In [ ]:
```
import re
import matplotlib.pyplot as plt

def count_instructions(file_name):
    with open(file_name, 'r') as f:
        lines = f.readlines()
        return len(lines)

if __name__ == '__main__':
    file_name = 'x86_64_gcc.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')


file = open('x86_64_gcc.asm','r')
inst_list=[]
for instruction in file:
        instruction = instruction.strip()

        if not re.search("^\.",instruction) and not re.search("^@.",instruction) and r
                inst_list.append(instruction)
mode_count = {"Scaled Index with disp":0,"Scaled Index":0,"Displacement":0,"Register I

for instruction  in inst_list:
        if re.search("\$.*,%",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search("0x.*\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index with disp')
                mode_count["Scaled Index with disp"]+=1

        elif re.search("\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index')
                mode_count["Scaled Index"]+=1

        elif re.search("0x.*\(.*\)",instruction) :
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1

        elif re.search("\(.*\),%",instruction) or re.search("\$",instruction):
                #print(instruction,'\t\tRegister Indirect')
                mode_count["Register Indirect"]+=1


        elif re.search("%.*,%",instruction):
                #print(instruction,'\t\Register Direct')
                mode_count["Register Direct"]+=1


        else:
                #print(instruction,'\t\tOthers')
                mode_count["Others"]+=1
```

```
for mode,count in mode_count.items():
        print(mode,'\t',count)


plt.bar(range(len(mode_count)), list(mode_count.values()), align='center')
plt.xticks(range(len(mode_count)), list(mode_count.keys()))
plt.show()
```

```
Number of instructions: 140248
Scaled Index with disp   4168
Scaled Index     1565
Displacement     33102
Register Indirect        3093
Immediate        19458
Register Direct          30937
Others   47787
```

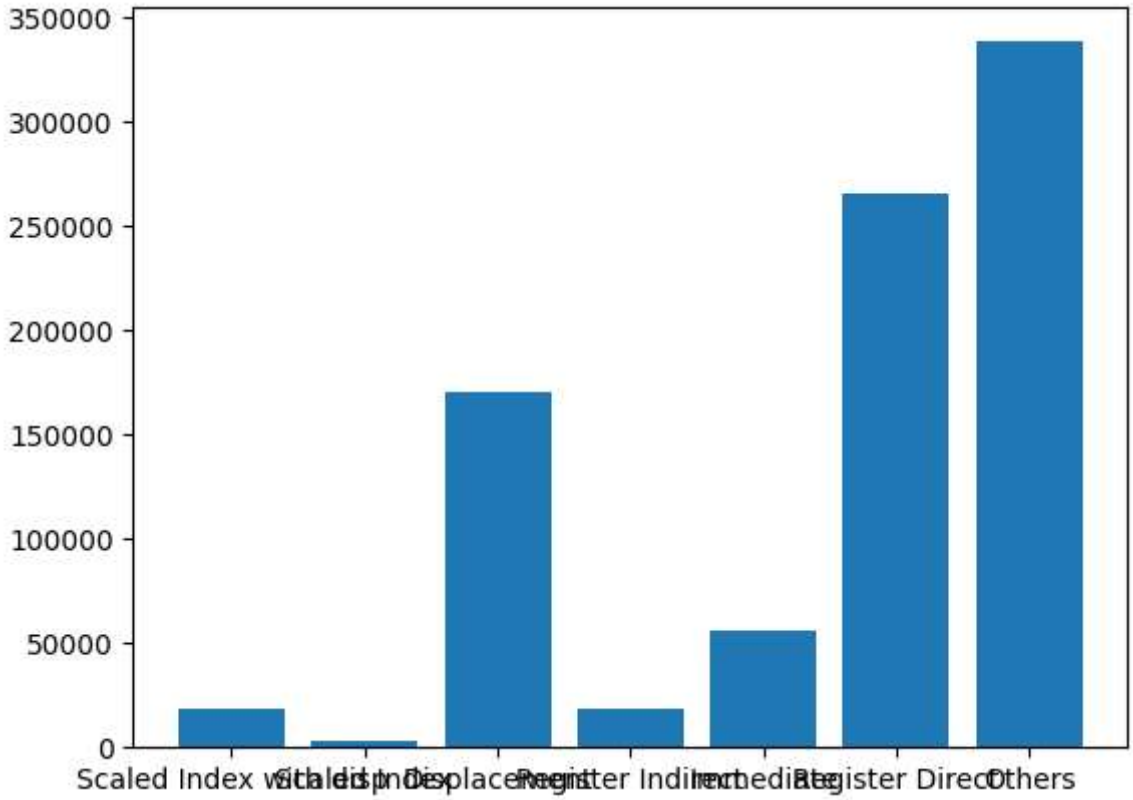


In [ ]:

```
import re
import matplotlib.pyplot as plt

def count_instructions(file_name):
    with open(file_name, 'r') as f:
        lines = f.readlines()
        return len(lines)

if __name__ == '__main__':
    file_name = 'x86_64_gimp.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')


file = open('x86_64_gimp.asm','r')
inst_list=[]
for instruction in file:
```

```
        instruction = instruction.strip()

        if not re.search("^\.",instruction) and not re.search("^@.",instruction) and r
                inst_list.append(instruction)
mode_count = {"Scaled Index with disp":0,"Scaled Index":0,"Displacement":0,"Register I

for instruction  in inst_list:
        if re.search("\$.*,%",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search("0x.*\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index with disp')
                mode_count["Scaled Index with disp"]+=1

        elif re.search("\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index')
                mode_count["Scaled Index"]+=1

        elif re.search("0x.*\(.*\)",instruction) :
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1

        elif re.search("\(.*\),%",instruction) or re.search("\$",instruction):
                #print(instruction,'\t\tRegister Indirect')
                mode_count["Register Indirect"]+=1


        elif re.search("%.*,%",instruction):
                #print(instruction,'\t\Register Direct')
                mode_count["Register Direct"]+=1


        else:
                #print(instruction,'\t\tOthers')
                mode_count["Others"]+=1

for mode,count in mode_count.items():
        print(mode,'\t',count)


plt.bar(range(len(mode_count)), list(mode_count.values()), align='center')
plt.xticks(range(len(mode_count)), list(mode_count.keys()))
plt.show()
```

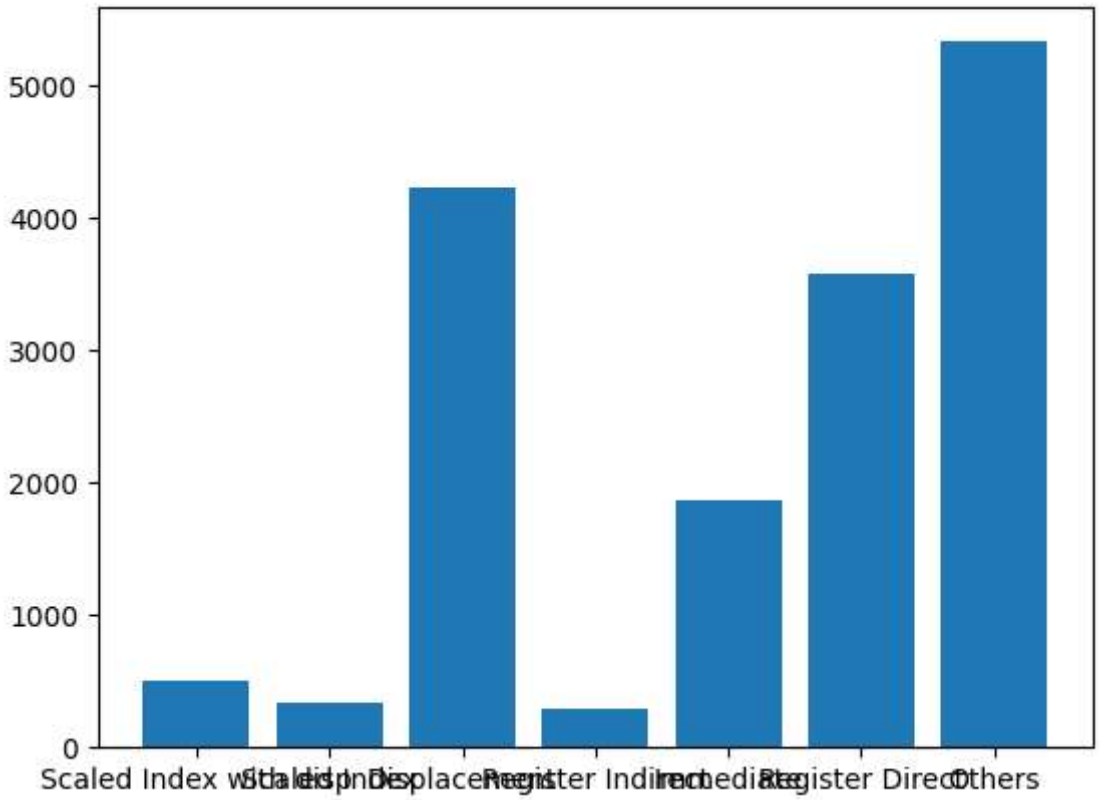
```
Number of instructions: 877881
Scaled Index with disp   18339
Scaled Index     2696
Displacement     170712
Register Indirect        18355
Immediate        55715
Register Direct          266243
Others   338634
```



In [ ]:
```
import re
import matplotlib.pyplot as plt

def count_instructions(file_name):
    with open(file_name, 'r') as f:
        lines = f.readlines()
        return len(lines)

if __name__ == '__main__':
    file_name = 'x86_64_gzip.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')


file = open('x86_64_gzip.asm','r')
inst_list=[]
for instruction in file:
        instruction = instruction.strip()

        if not re.search("^\.",instruction) and not re.search("^@.",instruction) and r
                inst_list.append(instruction)
mode_count = {"Scaled Index with disp":0,"Scaled Index":0,"Displacement":0,"Register I

for instruction  in inst_list:
        if re.search("\$.*,%",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search("0x.*\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index with disp')
                mode_count["Scaled Index with disp"]+=1

        elif re.search("\(.*,.*,.*\)",instruction):
```

```
                #print(instruction,'\t\Scaled Index')
                mode_count["Scaled Index"]+=1

        elif re.search("0x.*\(.*\)",instruction) :
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1

        elif re.search("\(.*\),%",instruction) or re.search("\$",instruction):
                #print(instruction,'\t\tRegister Indirect')
                mode_count["Register Indirect"]+=1


        elif re.search("%.*,%",instruction):
                #print(instruction,'\t\Register Direct')
                mode_count["Register Direct"]+=1


        else:
                #print(instruction,'\t\tOthers')
                mode_count["Others"]+=1

for mode,count in mode_count.items():
        print(mode,'\t',count)


plt.bar(range(len(mode_count)), list(mode_count.values()), align='center')
plt.xticks(range(len(mode_count)), list(mode_count.keys()))
plt.show()
```

```
Number of instructions: 16205
Scaled Index with disp   507
Scaled Index     334
Displacement     4228
Register Indirect        288
Immediate        1870
Register Direct          3575
Others   5336
```

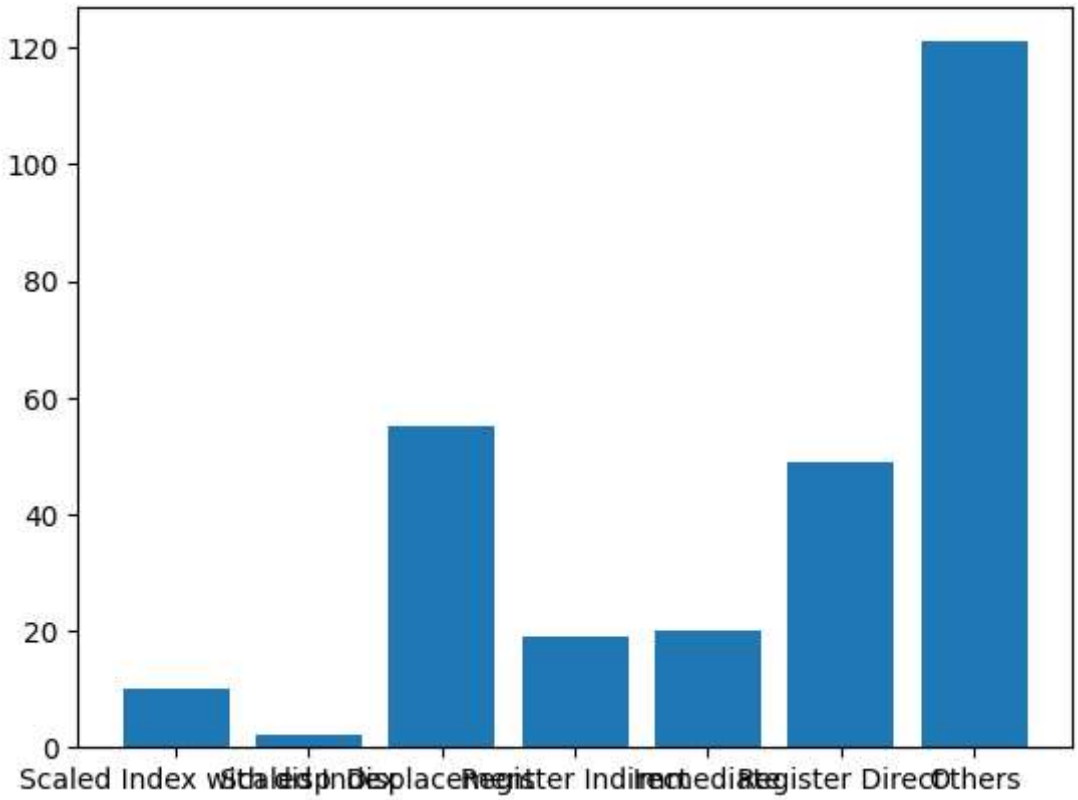


In [ ]:

```
import re
import matplotlib.pyplot as plt

def count_instructions(file_name):
    with open(file_name, 'r') as f:
        lines = f.readlines()
        return len(lines)

if __name__ == '__main__':
    file_name = 'x86_64_javac.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')


file = open('x86_64_javac.asm','r')
inst_list=[]
for instruction in file:
        instruction = instruction.strip()

        if not re.search("^\.",instruction) and not re.search("^@.",instruction) and r
                inst_list.append(instruction)
mode_count = {"Scaled Index with disp":0,"Scaled Index":0,"Displacement":0,"Register I

for instruction  in inst_list:
        if re.search("\$.*,%",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search("0x.*\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index with disp')
                mode_count["Scaled Index with disp"]+=1

        elif re.search("\(.*,.*,.*\)",instruction):
```

```
                #print(instruction,'\t\Scaled Index')
                mode_count["Scaled Index"]+=1

        elif re.search("0x.*\(.*\)",instruction) :
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1

        elif re.search("\(.*\),%",instruction) or re.search("\$",instruction):
                #print(instruction,'\t\tRegister Indirect')
                mode_count["Register Indirect"]+=1


        elif re.search("%.*,%",instruction):
                #print(instruction,'\t\Register Direct')
                mode_count["Register Direct"]+=1


        else:
                #print(instruction,'\t\tOthers')
                mode_count["Others"]+=1

for mode,count in mode_count.items():
        print(mode,'\t',count)


plt.bar(range(len(mode_count)), list(mode_count.values()), align='center')
plt.xticks(range(len(mode_count)), list(mode_count.keys()))
plt.show()
```

```
Number of instructions: 293
Scaled Index with disp   10
Scaled Index     2
Displacement     55
Register Indirect        19
Immediate        20
Register Direct          49
Others   121
```

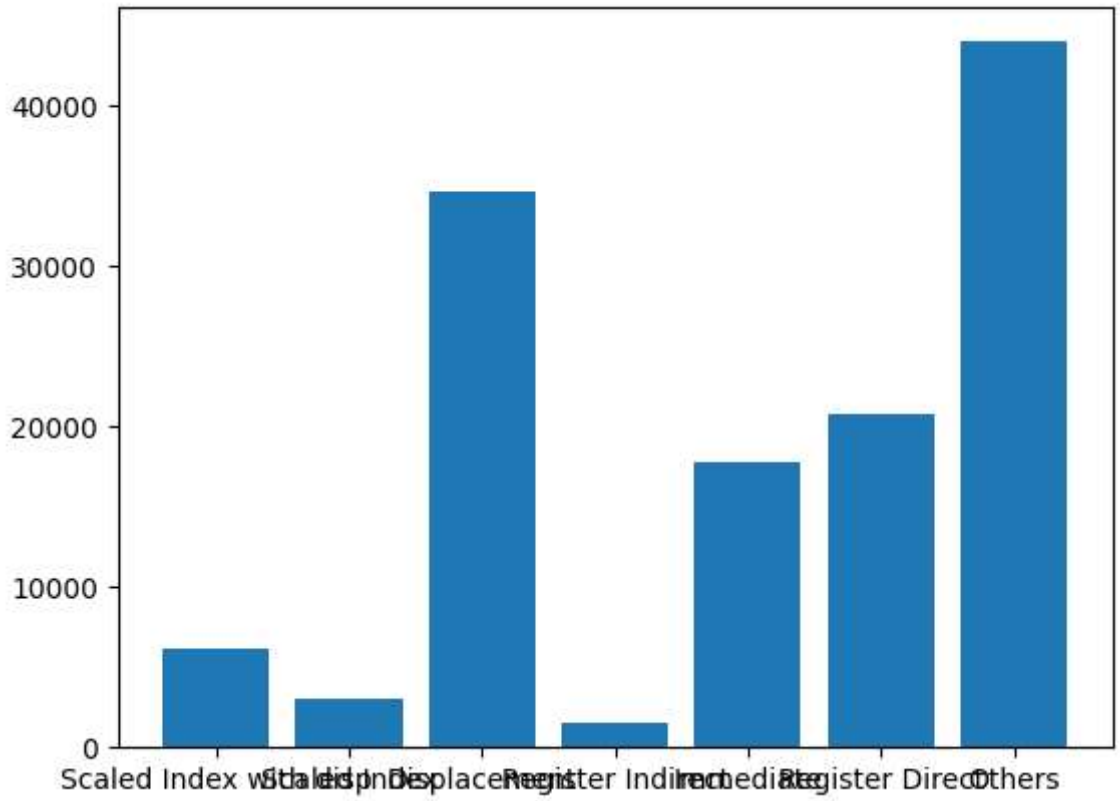


In [ ]:

```
import re
import matplotlib.pyplot as plt

def count_instructions(file_name):
    with open(file_name, 'r') as f:
        lines = f.readlines()
        return len(lines)

if __name__ == '__main__':
    file_name = 'x86_64_latex.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')


file = open('x86_64_latex.asm','r')
inst_list=[]
for instruction in file:
        instruction = instruction.strip()

        if not re.search("^\.",instruction) and not re.search("^@.",instruction) and r
                inst_list.append(instruction)
mode_count = {"Scaled Index with disp":0,"Scaled Index":0,"Displacement":0,"Register I

for instruction  in inst_list:
        if re.search("\$.*,%",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search("0x.*\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index with disp')
                mode_count["Scaled Index with disp"]+=1

        elif re.search("\(.*,.*,.*\)",instruction):
```

```
                #print(instruction,'\t\Scaled Index')
                mode_count["Scaled Index"]+=1

        elif re.search("0x.*\(.*\)",instruction) :
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1

        elif re.search("\(.*\),%",instruction) or re.search("\$",instruction):
                #print(instruction,'\t\tRegister Indirect')
                mode_count["Register Indirect"]+=1


        elif re.search("%.*,%",instruction):
                #print(instruction,'\t\Register Direct')
                mode_count["Register Direct"]+=1


        else:
                #print(instruction,'\t\tOthers')
                mode_count["Others"]+=1

for mode,count in mode_count.items():
        print(mode,'\t',count)


plt.bar(range(len(mode_count)), list(mode_count.values()), align='center')
plt.xticks(range(len(mode_count)), list(mode_count.keys()))
plt.show()
```

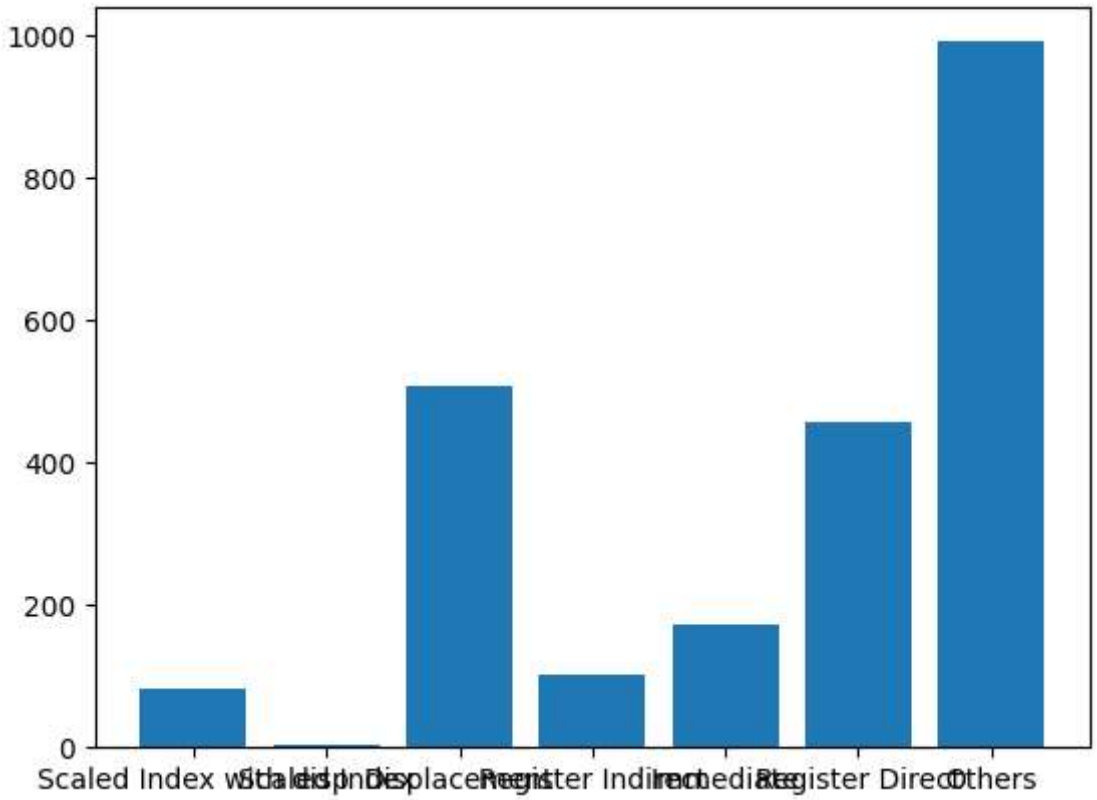
```
Number of instructions: 128067
Scaled Index with disp   6162
Scaled Index     3060
Displacement     34602
Register Indirect        1551
Immediate        17773
Register Direct          20806
Others   43925
```



In [ ]:
```
import re
import matplotlib.pyplot as plt

def count_instructions(file_name):
    with open(file_name, 'r') as f:
        lines = f.readlines()
        return len(lines)

if __name__ == '__main__':
    file_name = 'x86_64_octave.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')


file = open('x86_64_octave.asm','r')
inst_list=[]
for instruction in file:
        instruction = instruction.strip()

        if not re.search("^\.",instruction) and not re.search("^@.",instruction) and r
                inst_list.append(instruction)
mode_count = {"Scaled Index with disp":0,"Scaled Index":0,"Displacement":0,"Register I

for instruction  in inst_list:
        if re.search("\$.*,%",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search("0x.*\(.*,.*,.*\)",instruction):
                #print(instruction,'\t\Scaled Index with disp')
                mode_count["Scaled Index with disp"]+=1

        elif re.search("\(.*,.*,.*\)",instruction):
```

```
                #print(instruction,'\t\Scaled Index')
                mode_count["Scaled Index"]+=1

        elif re.search("0x.*\(.*\)",instruction) :
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1

        elif re.search("\(.*\),%",instruction) or re.search("\$",instruction):
                #print(instruction,'\t\tRegister Indirect')
                mode_count["Register Indirect"]+=1


        elif re.search("%.*,%",instruction):
                #print(instruction,'\t\Register Direct')
                mode_count["Register Direct"]+=1


        else:
                #print(instruction,'\t\tOthers')
                mode_count["Others"]+=1

for mode,count in mode_count.items():
        print(mode,'\t',count)


plt.bar(range(len(mode_count)), list(mode_count.values()), align='center')
plt.xticks(range(len(mode_count)), list(mode_count.keys()))
plt.show()
```

```
Number of instructions: 2373
Scaled Index with disp   81
Scaled Index     3
Displacement     507
Register Indirect        103
Immediate        171
Register Direct          457
Others   990
```


1000
800
600
400
200
0
Scaled Index w
Displacement
Register Indirect
Register Direct
Others