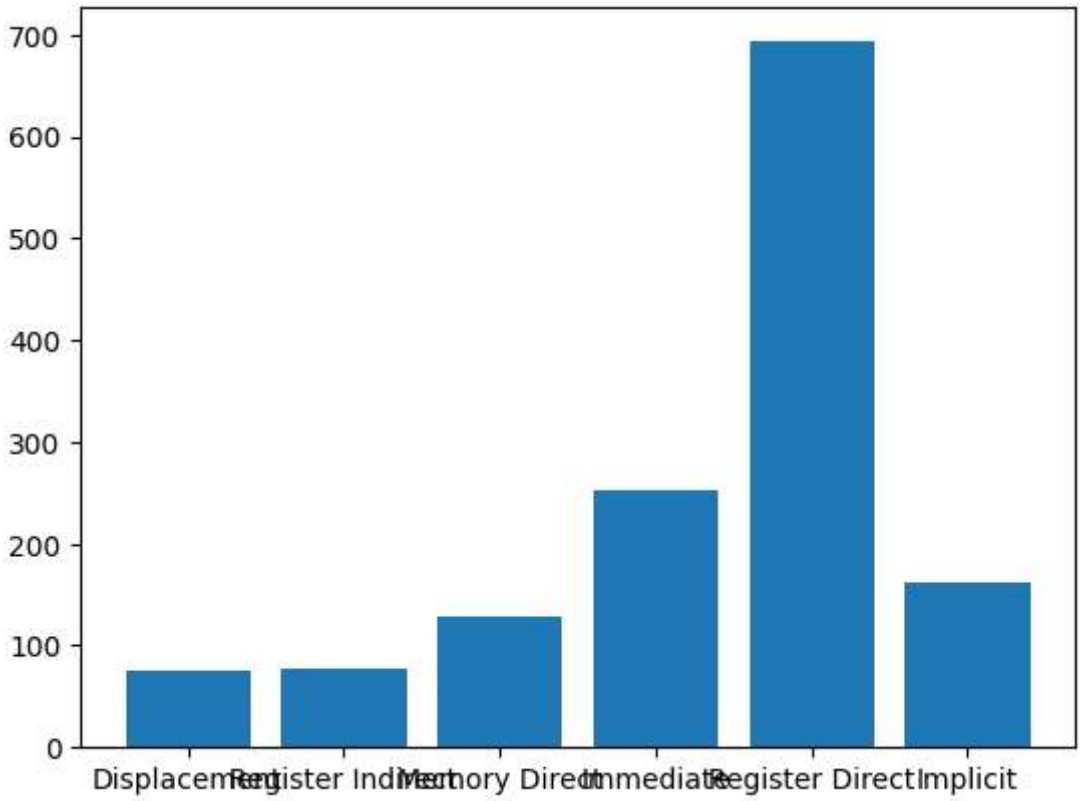
Niraj Nagrale CS22M109

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 = 'arm_c_javac.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')

file = open('arm_c_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 = {"Displacement":0,"Register Indirect":0,"Memory Direct":0,"Immediate":0,'

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

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

        elif re.search(".*\{.*}",instruction) or re.search("^b",instruction) or re.sea
                #print(instruction,'\t\tMemory Direct')
                mode_count["Memory Direct"]+=1

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

        elif re.search(".*r\d\, r\d",instruction) or re.search(".*r\d\, r\d\, r\d",ins
                #print(instruction,'\t\tRegister Direct')
                mode_count["Register Direct"]+=1

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

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

#Printing bar graph
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: 1488
Displacement     76
Register Indirect        78
Memory Direct    129
Immediate        253
Register Direct          693
Implicit         162
```

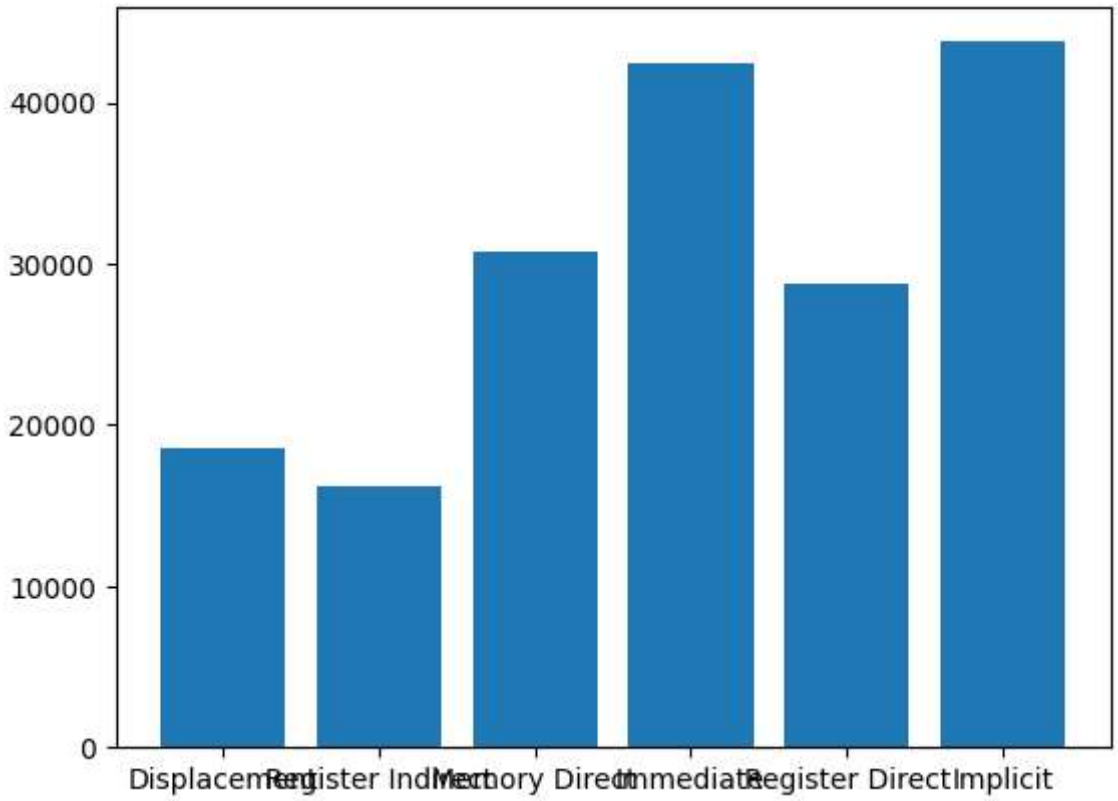


In [ ]:

```python
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 = 'arm_c_gcc.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')

file = open('arm_c_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 = {"Displacement":0,"Register Indirect":0,"Memory Direct":0,"Immediate":0,'

for instruction  in inst_list:
        if re.search(".*\[.*\#.*]",instruction):
                #print(instruction,'\t\tDisplacement')
                mode_count["Displacement"]+=1
```

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

        elif re.search(".*\{.*}",instruction) or re.search("^b",instruction) or re.sea
                #print(instruction,'\t\tMemory Direct')
                mode_count["Memory Direct"]+=1

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

        elif re.search(".*r\d\, r\d",instruction) or re.search(".*r\d\, r\d\, r\d",ins
                #print(instruction,'\t\tRegister Direct')
                mode_count["Register Direct"]+=1

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

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

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

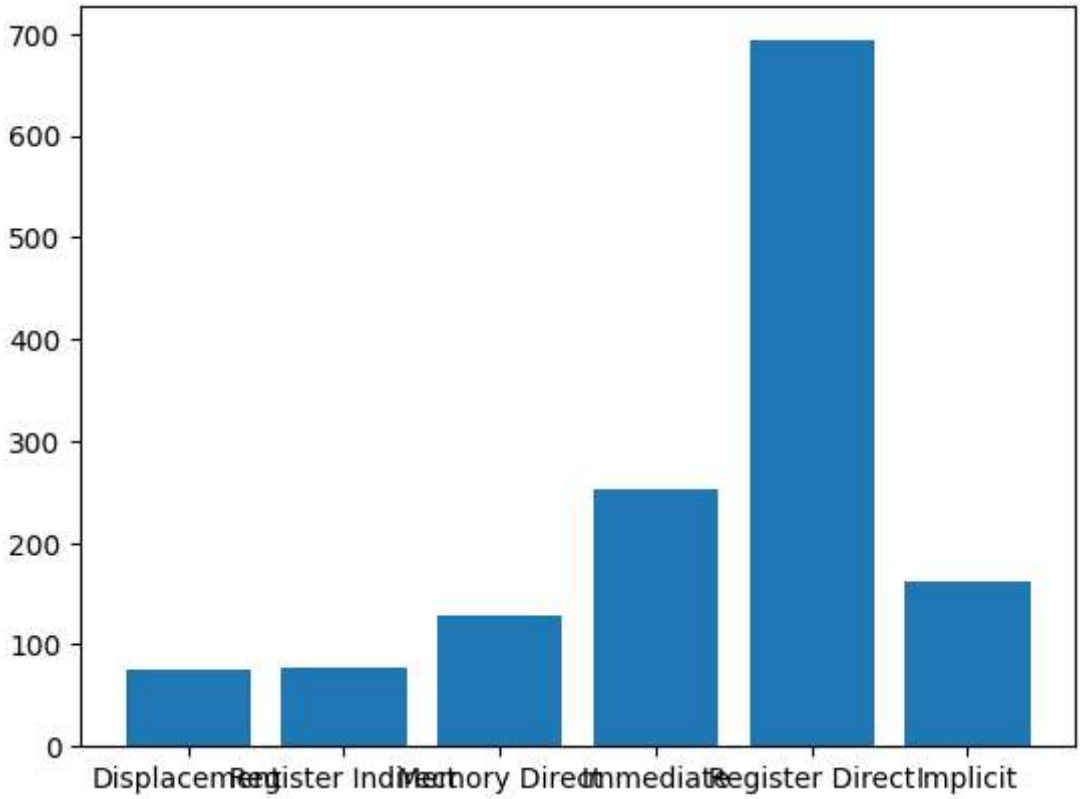
```
Number of instructions: 185863
Displacement     18592
Register Indirect        16211
Memory Direct    30741
Immediate        42393
Register Direct          28785
Implicit         43758
```



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 = 'arm_c_gimp.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')

file = open('arm_c_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 = {"Displacement":0,"Register Indirect":0,"Memory Direct":0,"Immediate":0,'

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

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

        elif re.search(".*\{.*}",instruction) or re.search("^b",instruction) or re.sea
                #print(instruction,'\t\tMemory Direct')
                mode_count["Memory Direct"]+=1
```

```
        elif re.search(".*\#\d.*",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search(".*r\d\, r\d",instruction) or re.search(".*r\d\, r\d\, r\d",ins
                #print(instruction,'\t\tRegister Direct')
                mode_count["Register Direct"]+=1

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

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

#Printing bar graph
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: 1488
Displacement     76
Register Indirect        78
Memory Direct    129
Immediate        253
Register Direct          693
Implicit         162
```

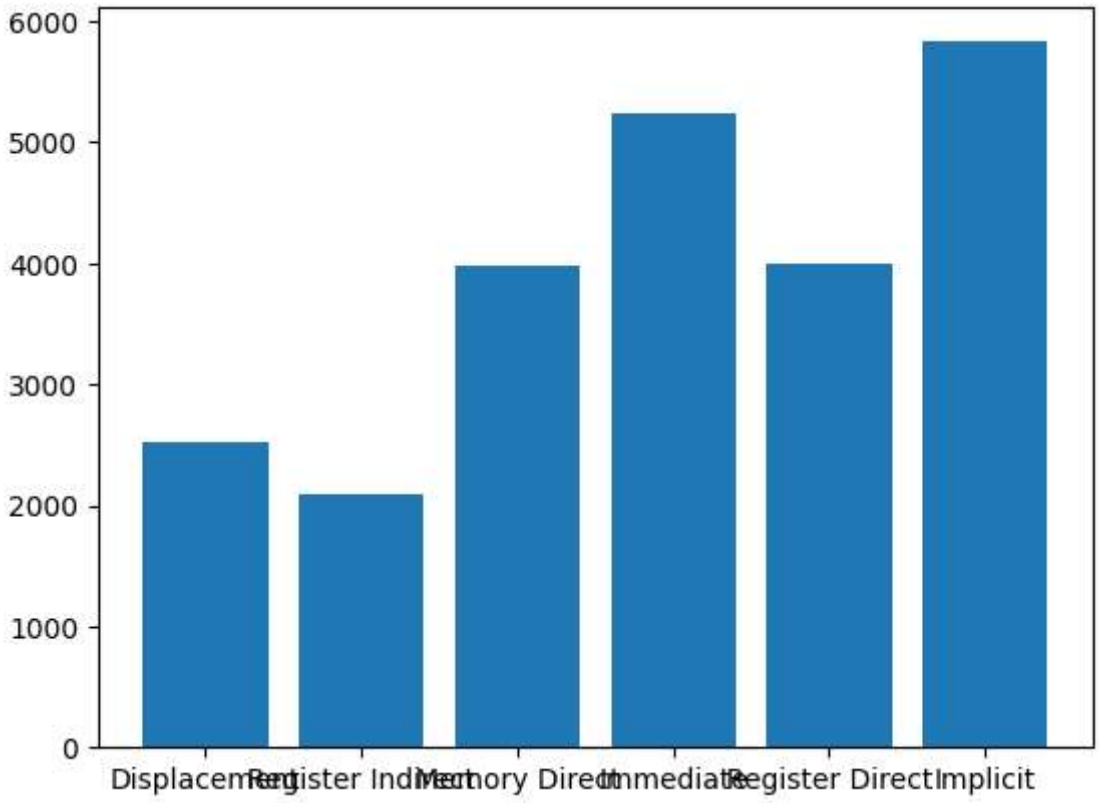


In [ ]:

```
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 = 'arm_c_gzip.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')

file = open('arm_c_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 = {"Displacement":0,"Register Indirect":0,"Memory Direct":0,"Immediate":0,'

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

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

        elif re.search(".*\{.*}",instruction) or re.search("^b",instruction) or re.sea
                #print(instruction,'\t\tMemory Direct')
                mode_count["Memory Direct"]+=1

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

        elif re.search(".*r\d\, r\d",instruction) or re.search(".*r\d\, r\d\, r\d",ins
                #print(instruction,'\t\tRegister Direct')
                mode_count["Register Direct"]+=1

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

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

#Printing bar graph
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: 23977
Displacement     2516
Register Indirect        2086
Memory Direct    3988
Immediate        5239
Register Direct          4002
Implicit         5832
```

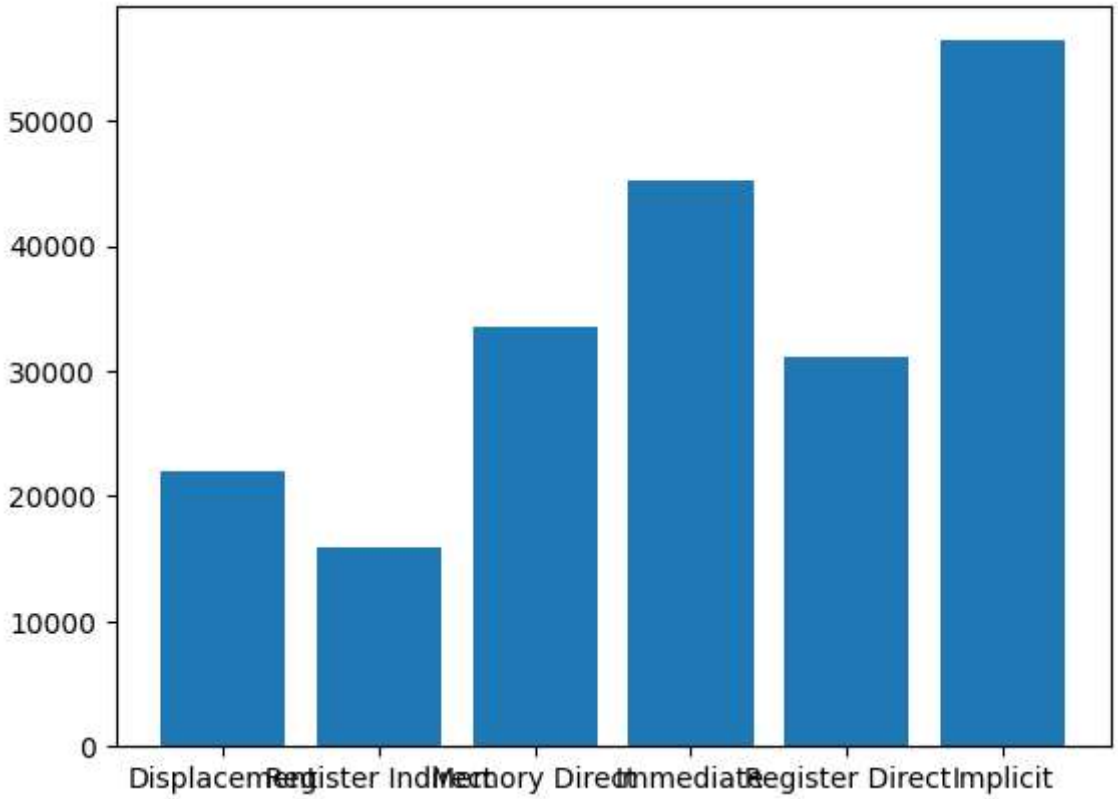


In [ ]:

```
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 = 'arm_c_latex.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')

file = open('arm_c_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 = {"Displacement":0,"Register Indirect":0,"Memory Direct":0,"Immediate":0,'

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

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

        elif re.search(".*\{.*}",instruction) or re.search("^b",instruction) or re.sea
                #print(instruction,'\t\tMemory Direct')
                mode_count["Memory Direct"]+=1
```

```
        elif re.search(".*\#\d.*",instruction):
                #print(instruction,'\t\tImmediate')
                mode_count["Immediate"]+=1

        elif re.search(".*r\d\, r\d",instruction) or re.search(".*r\d\, r\d\, r\d",ins
                #print(instruction,'\t\tRegister Direct')
                mode_count["Register Direct"]+=1

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

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

#Printing bar graph
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: 204806
Displacement     21942
Register Indirect        15971
Memory Direct    33524
Immediate        45163
Register Direct          31064
Implicit         56410
```

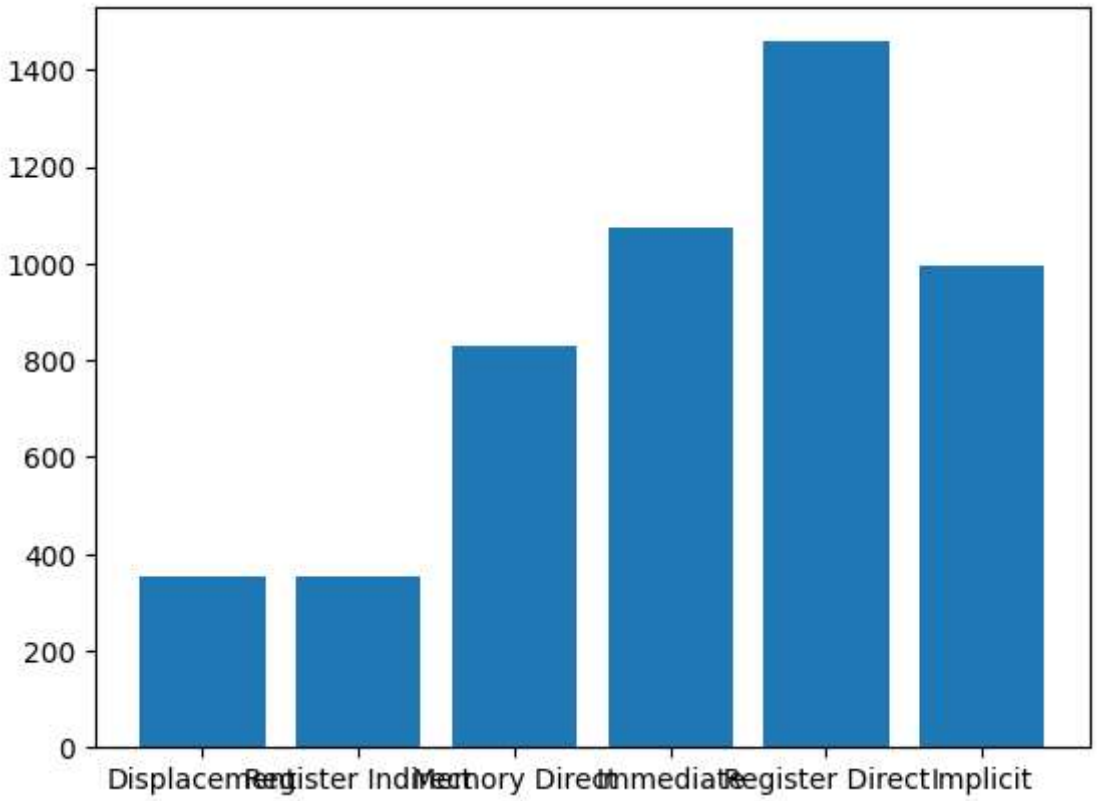


In [ ]:

```
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 = 'arm_c_octave.asm'
    instructions = count_instructions(file_name)
    print(f'Number of instructions: {instructions}')

file = open('arm_c_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 = {"Displacement":0,"Register Indirect":0,"Memory Direct":0,"Immediate":0,'

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

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

        elif re.search(".*\{.*}",instruction) or re.search("^b",instruction) or re.sea
                #print(instruction,'\t\tMemory Direct')
                mode_count["Memory Direct"]+=1

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

        elif re.search(".*r\d\, r\d",instruction) or re.search(".*r\d\, r\d\, r\d",ins
                #print(instruction,'\t\tRegister Direct')
                mode_count["Register Direct"]+=1

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

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

#Printing bar graph
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: 5275
Displacement     355
Register Indirect        352
Memory Direct    828
Immediate        1072
Register Direct          1458
Implicit         995
```


1400
1200
1000
800
600
400
200
0
Displacement
Register Indirect
Memory Direct
Immediate
Register Direct
Implicit