ExperimentNo.:-2

WriteaprogramtoimplementHuffmanEncodingusingagreedystrategy.

Source Code:-

In[1]:

```
importheapq

classNode:
    definit(self, freq, symbol, left=None,
        right=None):self.freq=freq
        self.symbol=symbols
        elf.left=left
        self.right=right
        self.huff=""

    deflt(self,other):
        returnself.freq<other.freq

defprintNodes(node,val=""):newval=v
    al+node.huff
    ifnode.leftornode.right:
        ifnode.left:
            printNodes(node.left,newval)
        ifnode.right:
            printNodes(node.right,newval)
    else:
        print(f"{node.symbol}->{newval}")
        encoded_lengths[node.symbol]=len(newval)

#Gettinguserinputforcharactersandtheirfrequencies
num_chars=int(input("Enternumberofcharacters:"))chars
=[]
freqs=[]

foriinrange(num_chars):
    char=input(f"Entercharacter{i+1}:")
    freq=int(input(f"Enterfrequencyofcharacter{char}:"))chars.
    append(char)
    freqs.append(freq)node

s=[]

foriinrange(len(chars)):
    heapq.heappush(nodes,Node(freqs[i],chars[i]))

whilelen(nodes)>1:
    left =
    heapq.heappop(nodes)right=heap
    q.heappop(nodes)
```

```
    left.huff="0"
    right.huff="1"
    newnode=Node(left.freq+right.freq,left.symbol+right.symbol,left,right)heapq.he
    appush(nodes,newnode)

#Calculatingtotalsizebeforeencoding
total_size_before=sum(freqs)*8

#Printingthenodesandcalculatingencodedlengths
encoded_lengths={}pr
intNodes(nodes[0])

#Calculatingtotalsizeafterencoding
total_size_after=sum(freqs[i]*encoded_lengths[chars[i]]foriinrange(num_chars))

#CalculatingEncodedDataRepresentation
characters=num_chars*8fre
quency=sum(freqs)
encoded_data_representation=characters+frequency+total_size_after

print("\nTotalsizebeforeencoding:",total_size_before,"bits")print("Totalsizeaf
terencoding:",total_size_after,"bits")
print("EncodedDataRepresentation:",encoded_data_representation,"bits")
```

```
Enter numberof  characters:4
Enter character 1:B
Enter frequency ofcharacter  B: 1
Enter character 2:C
Enter frequency ofcharacter  C: 6
Enter character 3:A
Enter frequency ofcharacter  A: 5
Enter character 4:D
Enter frequency ofcharacter  D: 3
 C->  0
 B->  100
 D->  101
 A->  11

Totalsizebeforeencoding:120bitsTota
lsizeafterencoding:28bits
EncodedDataRepresentation:75bits
```

In[]: