```
def valid_ip(ip_address):
 element = ip_address.split('.')
 print(element)
 if len(element) != 4:
 return False
 for elem in element:
 if not elem.isdigit():
 return False
 num = int(elem)
 if num < 0 or num > 255:
 return False
 return True
def get_ip_class(ip_address):
 components = ip_address.split('.')
 first_octet = int(components[0])
 if 1 <= first_octet <= 126:
 return "class A \nSubnetting mask is 255.0.0.0"
 elif 128 <= first_octet <= 191:
 return "class B \nSubnetting mask is 255.255.0.0"
 elif 192 <= first_octet <= 223:
 return "class C \nSubnetting mask is 255.255.255.0"
 elif 224 <= first_octet <= 239:
 return "class D \nSubnetting mask is not equipped with any subnet mask"
 elif 240 <= first_octet <= 255:
 return "class E \nSubnetting mask is not equipped with any subnet mask"
 elif first_octet == 0 or 127:
 return "network address"
 else:
 return "unknown"
input = input("Enter your ip address: ")
if valid_ip(input):
 print("Valid ip address")
 ip_class = get_ip_class(input)
 print("ip address belongs to ", ip_class)
 print("")
else:
 print("Invalid ip address")
```