```
class HashMap:
    def __init__(self):
        self.size = 16
        self.map = [None] * self.size

    def _get_hash(self, key):
        return hash(key) % self.size

    def add(self, key, value):
        key_hash = self._get_hash(key)
        key_value = [key, value]

        if self.map[key_hash] is None:
            self.map[key_hash] = list([key_value])
            return True
        else:
            for pair in self.map[key_hash]:
                if pair[0] == key:
                    pair[1] = value
                    return True
            self.map[key_hash].append(key_value)
            return True

    def get(self, key):
        key_hash = self._get_hash(key)
        if self.map[key_hash] is not None:
            for pair in self.map[key_hash]:
                if pair[0] == key:
                    return pair[1]
        return None

    def delete(self, key):
        key_hash = self._get_hash(key)
        if self.map[key_hash] is None:
            return False
        for i in range(0, len(self.map[key_hash])):
            if self.map[key_hash][i][0] == key:
                self.map[key_hash].pop(i)
                return True
        return False

    def print(self):
        for item in self.map:
            if item is not None:
```

```
        print(str(item))
```