|  |  |
| --- | --- |
| **Highest Frequency Char in C++** | |
| #include <iostream>  #include <unordered\_map>  #include <string>  using namespace std;  char getHighestFrequencyChar(string str) {  unordered\_map<char, int> hm; // HashMap to store character frequencies  // Count frequencies of characters in the string  for (char ch : str) {  hm[ch]++;  }  char mfc = str[0]; // Initialize most frequent character with the first character  // Find the character with the highest frequency  for (auto it = hm.begin(); it != hm.end(); ++it) {  if (it->second > hm[mfc]) {  mfc = it->first;  }  }  return mfc;  }  int main() {  string str = "zmszeqxllzvheqwrofgcuntypejcxovtaqbnqyqlmrwitc";  char highestFreqChar = getHighestFrequencyChar(str);  cout << highestFreqChar << endl;  return 0;  } | ****Input**** String: "zmszeqxllzvheqwrofgcuntypejcxovtaqbnqyqlmrwitc" ****Step 1: Count Character Frequencies**** We iterate through the string str and populate the unordered\_map (hm) with the count of each character. ****Character Frequency Count:****  | **Character** | **Count** | | --- | --- | | z | 3 | | m | 3 | | s | 2 | | e | 4 | | q | 4 | | x | 2 | | l | 3 | | v | 2 | | h | 1 | | w | 2 | | r | 2 | | o | 2 | | f | 1 | | g | 1 | | c | 2 | | u | 1 | | n | 2 | | t | 2 | | y | 3 | | p | 1 | | j | 1 | | a | 1 | | b | 1 | | i | 1 |  ****Step 2: Find the Character with the Highest Frequency**** We iterate through the unordered\_map (hm) and keep track of the character with the maximum frequency (mfc). Initially, mfc is set to the first character of the string, z. ****Iteration Over HashMap:****  | **Current Character** | **Frequency** | **hm[mfc]** | **Update mfc?** | **Updated mfc** | | --- | --- | --- | --- | --- | | z | 3 | 3 | No | z | | m | 3 | 3 | No | z | | s | 2 | 3 | No | z | | e | 4 | 3 | Yes | e | | q | 4 | 4 | No | e | | x | 2 | 4 | No | e | | l | 3 | 4 | No | e | | v | 2 | 4 | No | e | | h | 1 | 4 | No | e | | w | 2 | 4 | No | e | | r | 2 | 4 | No | e | | o | 2 | 4 | No | e | | f | 1 | 4 | No | e | | g | 1 | 4 | No | e | | c | 2 | 4 | No | e | | u | 1 | 4 | No | e | | n | 2 | 4 | No | e | | t | 2 | 4 | No | e | | y | 3 | 4 | No | e | | p | 1 | 4 | No | e | | j | 1 | 4 | No | e | | a | 1 | 4 | No | e | | b | 1 | 4 | No | e | | i | 1 | 4 | No | e |  ****Step 3: Output**** The character with the highest frequency is **q**, appearing **4 times** in the string. ****Output**** |
| Output: q | |