Mirko has been moving up in the world of basketball. He started as a mere spectator, but has already reached the coveted position of the national team coach!

Mirco is now facing a difficult task: selecting five primary players for the upcoming match against Tajikistan. Since Mirko is incredibly lazy, he doesn't bother remembering players' names, let alone their actual skills. That's why he has settled on selecting five players who share the same first letter of their surnames, so that he can remember them easier. If there are no five players sharing the first letter of their surnames, Mirko will simply forfeit the game!

Find the first letters Mirko's players' surnames can begin with, or return"forfeit" if Mirko can't gather a team.

**Example:**

* For N = 6 and

players = ["michael","jordan","lebron",

"james","kobe","bryant"]`

the answer is Coach(N, players) = "forfeit".

* For N = 18 and

players = ["babic","keksic","boric","bukic",

"sarmic","balic","kruzic","hrenovkic",

"beslic","boksic","krafnic","pecivic",

"klavirkovic","kukumaric","sunkic","kolacic",

"kovacic","prijestolonasljednikovic"]

the answer is Coach(N, players) = "bk".

* **[input] integer N**
  + A positive integer, the number of players Mirko has. 1 ≤ N ≤ 150.
* **[input] array.string players**
  + Array contains N words at most30 characters long, consisting only of lowercase English letters, denoting players' surnames.
* **[output] string**
  + A sorted string of possible first letters, or "forfeit" if it's impossible to gather a team.

<https://codefights.com/challenge/wg9xv2BKTeyoebp9A>

static string Coach(int N, string[] players)

{

Dictionary<char, int> diccio = new Dictionary<char, int>();

for (int i = 0; i < players.Length; i++)

{

if (diccio.ContainsKey(players[i][0]))

{

diccio[players[i][0]]++;

}

else

{

diccio[players[i][0]] = 1;

}

}

string answer = "";

foreach (KeyValuePair<char, int> kvp in diccio)

{

if (kvp.Value >= 5)

{

answer += kvp.Key;

}

}

if (answer.Length == 0)

{

return "forfeit";

}

char[] sorted = answer.ToCharArray();

for (int i = 1; i < sorted.Length; i++)

{

int index = i;

while (index > 0 && sorted[index] < sorted[index - 1])

{

char temp = sorted[index];

sorted[index] = sorted[index - 1];

sorted[index - 1] = temp;

index--;

}

}

string s = "";

for (int i = 0; i < sorted.Length; i++)

{

s += sorted[i];

}

return s;

}