Hpc 1 bfs

```
#include <iostream>
#include <queue>

using namespace std;

struct Node {
  Node *left, *right;
  int data;
};

Node* insert(Node* root, int data) {
  if (!root) return new Node{NULL, NULL, data};
  queue<Node*> q{{root}};
  while (!q.empty()) {
     Node* curr = q.front(); q.pop();
     if (curr->left) q.push(curr->left);
     else return curr->left = new Node{NULL, NULL, data};
     if (curr->right) q.push(curr->right);
     else return curr->right = new Node{NULL, NULL, data};
  }
  return nullptr;
}

void bfs(Node* root) {
  if (!root) return;
  queue<Node*> q{{root}};
  while (!q.empty()) {
     Node* curr = q.front(); q.pop();
     cout << curr->data << " ";
     if (curr->left) q.push(curr->left);
     if (curr->right) q.push(curr->right);
  }
}

int main() {
  Node* root = nullptr;
  char ans;
  do {
     int data;
     cout << "Enter data: ";
     cin >> data;
     root = insert(root, data);
     cout << "Add another node? (y/n): ";
     cin >> ans;
  } while (ans == 'y' || ans == 'Y');
  cout << "BFS: ";
  bfs(root);
  return 0;
}
```