שאלה 1

1. Reference הוא בעצם הפניה שמשתמשים בה למשתנה כלשהו. כאשר מבצעים הפניה, אפשר להתייחס אליה בדיוק כמו שם שני למשתנה שהפנת אליו. כלומר, יהיה ניתן לשנות אותו ולערוך אותו בשם אחר ובכלל ב-scope אחר. Reference למשתנה מגדירים בעזרת סימן ה-&. לדוגמא:

int& foo = x

מעכשיו, המשתנה foo הוא הפניה למשתנה x. משמע ניתן לערוך את הערך של x בעזרת עריכה של המשתנה foo.

1. דבר ראשון, זה יותר יעיל. משום שכאשר שולחים לפונקציה reference למשתנה כלשהו. אנחנו בעיקרון שולחים מצביע לאובייקט שהוא כבר קיים. לעומת זאת, לשלוח על ידי ערך, דורש העתק הדבק של אותו אובייקט. זאת אומרת, יותר זמן וזיכרון דרושים בתהליך. דבר שני, מתייחסים להפניות כמו שתייחסים למשתנה רגיל, מאחר וזה שם שני למשתנה שניתן לגשת אליו אפילו בטווח scope שונה.
2. ההבדל ביניהם הוא שהפניה, אתה פונה אליה בדיוק כפי שאתה פונה למשתנה רגיל. כמו משתנה מסוג מספר שלם לדוגמא. בנוסף, כשאתה יוצר הפניה, אתה לא צריך לעשות משהו מיוחד כדי לקבל את כתובת הזיכרון של אותו משתנה, הקומפיילר עושה את זה בשבילך. לעומת מצביעים, reference אתה תמיד מאתחל את ההפניה, זאת אומרת שאתה אף פעם לא תצטרך לדאוג שזה יצביע ל-NULL או משהו לא תקין או שאתה לא אתחלת את ההפניה ושלהפניה הזיכרון לא יהיה מוקצה בצורה כזו או אחרת.
3. הקריאה מספר 1 תקינה, מאחר ו-result זה reference ל-y. אז ניתן להגיד שהפונקציה מבצעת פעולה ריבוע ושומרת את התוצאה בפרמטר השני שהפונקציה מקבלת שהוא reference. כלומר הפונקציה תבצע ריבוע של 3, ותשמור את התוצאה בפרמטר y.  
   הקריאה מספר 2 לא נכונה, משום שאנו שולחים כפרמטר שני מצביע למשתנה y, אבל, הפונקציה מצפה ל-reference של המשתנה y.  
   ולבסוף קריאה מספר 3 גם לא נכונה, מאחר והפרמטר השני מייצג מספר שלם. והפונקציה מצפה ל-reference למספר שלם. בנוסף למספר שלם אין הקצאת זיכרון משלו. הוא לא משתנה.
4. הבעיה בפונקציה זו, זה שהפונקציה מחזירה reference למשתנה לוקאלי שמאותחל בתוך הפונקציה. כלומר ברגע שמחזירים אליו reference, ההפניה מצביעה על מקום שכבר לא ניתן לגשת אליו. ומאחר והמשתנה x כבר לא בטווח ה-scope. כדי לתקן זו ניתן להחזיר מספר שלם במקום reference למספר שלם.
5. מאחר והפונקציה הזו מחזירה reference ולא מצביע, אז לא תהיה לנו דרך לשחרר את המשתנה הזה, מאחר ועכשיו הוא הפניה למשהו אחר. אנחנו בפונקציה מחזירים משתנה שמוקצה דינמית, כלומר מחזירים את הצביע שלו.